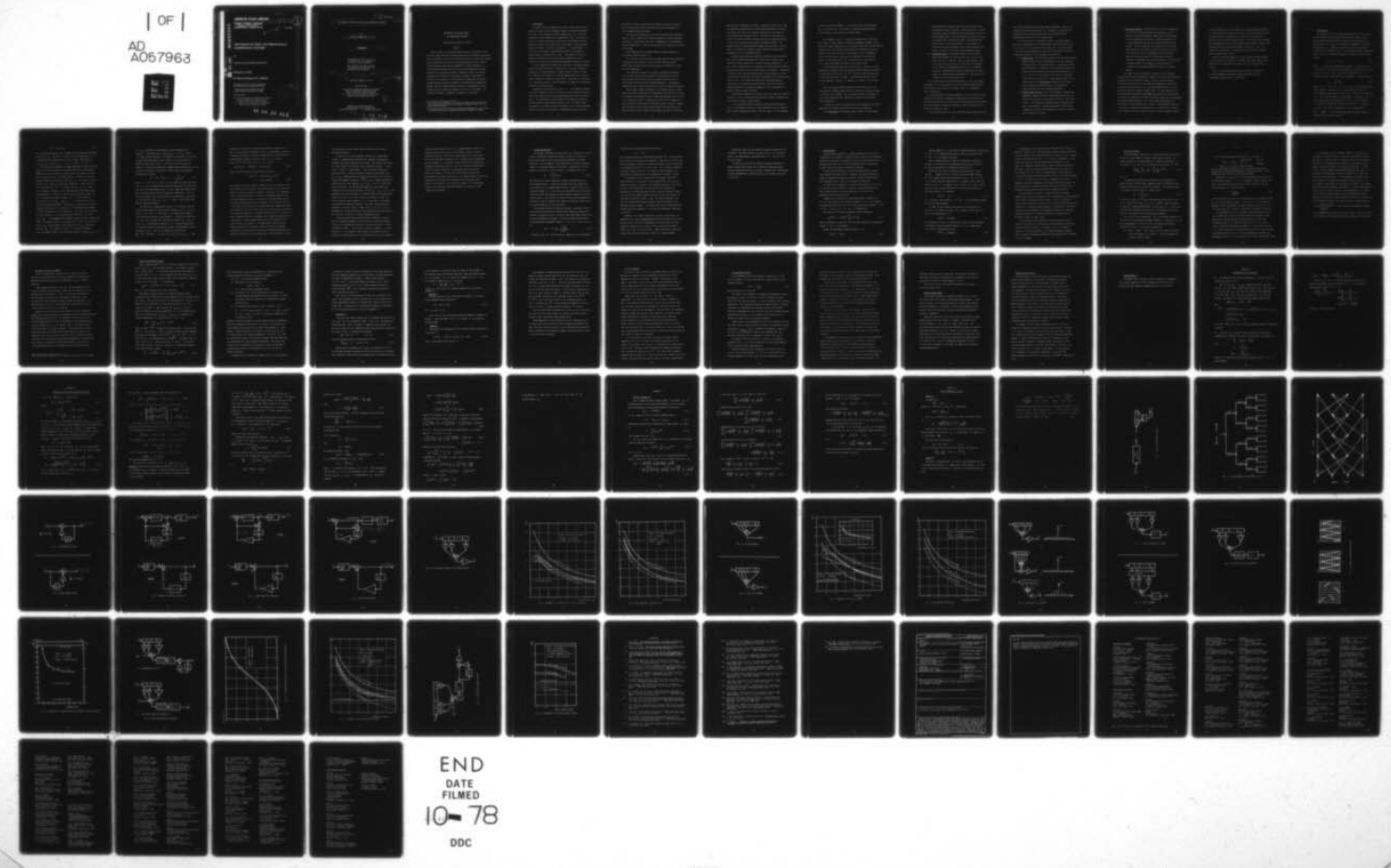
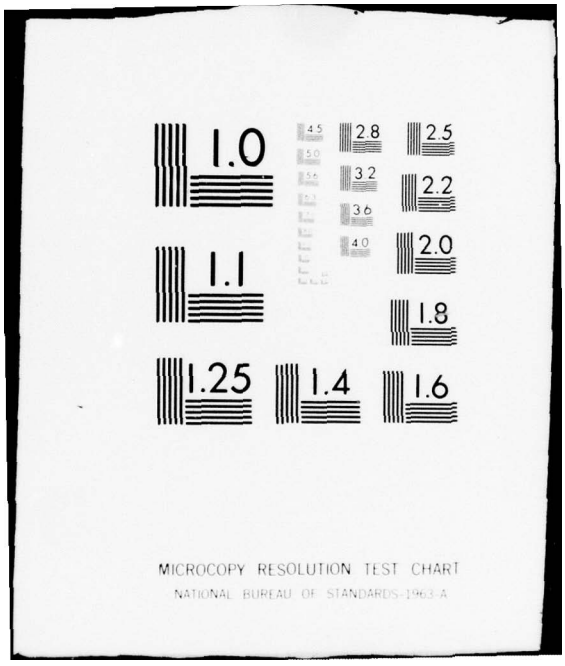
AD-A057 963 STANFORD UNIV CALIF STANFORD ELECTRONICS LABS F/G 9/2
THE DESIGN OF TREE AND TRELLIS DATA COMPRESSION SYSTEMS.(U)
FEB 78 J LINDE, R M GRAY F44620-73-C-0065
UNCLASSIFIED SU-SEL-78-004 NL

| OF |
AD
A057963



END
DATE
FILMED
10-78
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

INFORMATION SYSTEMS LABORATORY

STANFORD ELECTRONICS LABORATORIES
DEPARTMENT OF ELECTRICAL ENGINEERING
STANFORD UNIVERSITY · STANFORD, CA 94305

SEL-78-004

THE DESIGN OF TREE AND TRELLIS DATA COMPRESSION SYSTEMS

by

Joseph Linde and Robert M. Gray

February 1978

Technical Report No. 6504-2

Prepared under

U.S. Air Force Office of Scientific Research
Contract F44620-73-C-0065 and by the
Joint Services Electronics Program
(U.S. Army, U.S. Navy, and U.S. Air Force)
under Contract N00014-75-C-0601

(14) SU - SEL-78-004, TR-6504-2

(6) THE DESIGN OF TREE AND TRELLIS DATA COMPRESSION SYSTEMS

by

(10) Joseph Linde and Robert M. Gray

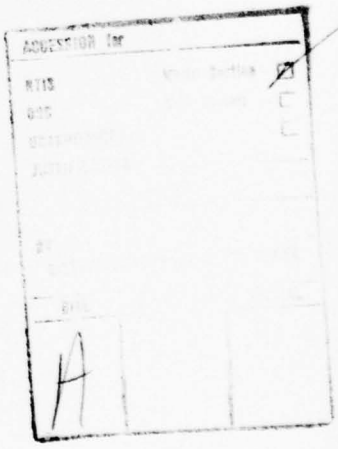
(11) February 1978 (12) 79 p.

(15) F44620-73-C-0065, N00014-75-C-0601

(9) Technical Report No. 6504-2

Prepared under

U.S. Air Force Office of Scientific Research
Contract F44620-73-C-0065 and by the
Joint Services Electronics Program
(U.S. Army, U.S. Navy, and U.S. Air Force)
under Contract N00014-75-C-0601



Information Systems Laboratory
Stanford Electronics Laboratories
Stanford University Stanford, California

THE DESIGN OF TREE AND TRELLIS DATA COMPRESSION SYSTEMS*

Joseph Linde and Robert M. Gray**

Abstract

Recent results in information theory promise the existence of tree and trellis data compression systems operating near the Shannon theoretical bound, but provide little indication of how actually to design such systems. We here present several intuitive design approaches and also a general design philosophy based upon the generation of "fake processes" i.e., finite entropy processes which are close (in the generalized Ornstein distance) to the process one wishes to compress. Most of the design procedures can be used for a wide class of sources. Performance is evaluated, via simulations, for memoryless, autoregressive and moving average Gaussian sources and compared to traditional Data Compression systems. The new schemes typically provide 1-2 dB improvement in performance over the traditional schemes at a rate of 1 bit/symbol. The inevitable increase in complexity is moderate in most cases.

---

* This research was supported by Air Force contract F44620-73-0065 and by the Joint Services Electronics Program at Stanford under U.S. Navy Grant N00013-67-A-0112.

** The authors are with the Electrical Engineering Department and Information Systems Laboratory, Stanford University, Stanford, CA 94305.

1. Introduction

A discrete time data compression or source coding system can be viewed as a means of first encoding a sequence of continuous-alphabet data $\{X_n\}$ into a sequence of binary symbols, say $\{Y_n\}$, and then decoding the binary symbols into a reproduction $\{\hat{X}_n\}$. The system is said to be a fixed rate system if each time a fixed number of source symbols are input to the encoder and a fixed number of binary encoded symbols are output. The rate of the system R (in bits/symbol) is the ratio of the number of encoded binary digits to the number of source symbols in a long period of time. The smaller the rate, the larger the "compression" since fewer binary symbols must then be transmitted in a given time and hence the required bandwidth is smaller.

In most communication systems the source has a continuous alphabet (usually the real line) and an infinite rate is therefore required to communicate the source perfectly. Usually however, one has a finite rate constraint due perhaps to a digital communication link, a finite capacity channel or a digital storage device. Even where no such finite rate constraint exists, it may be convenient to impose one to facilitate encryption for data security.

Regardless of the motivation, since $R = \infty$ is in general required for perfect reproduction, but a finite rate is used instead, distortion between the original source and delivered reproduction must inevitably result. The evaluation of lower bounds on the average distortions is the main concern of Rate-Distortion theory [1,2]. The goal of data compression is to design systems which operate close to these unbeatable bounds for a given rate constraint. Equivalently, one may be given a

distortion or fidelity constraint and attempt to minimize the rate to meet this requirement (lower rate means lower cost and complexity in later communication and storage).

The previous discussion also holds for continuous-time information sources, since a discrete-time source can always be obtained (possibly at the cost of additional distortion) via sampling or a transformation such as Karhunen-Loeve. Hence we have confine the discussion to discrete time systems.

Data compression has developed along two largely separate but occasionally connected paths [3].

(1) Several ad hoc but often quite good practical data compression systems have been developed in particular PCM, DPCM, predictive quantization and delta-modulation have become the workhorses of fixed rate data compression.

(2) Theoretical bounds on the optimal achievable performance (for a given rate) have been developed in information theory and it has been shown that certain classes of coding structures (such as block codes) can achieve nearly optimal performance. Unfortunately information theory rarely tells one how to find (or design) a good code.

The principle original application of information theory to data compression system design was the demonstration [4,5] that for very high rate systems and memoryless sources, simple PCM performed within .25 bit of the theoretical limit, indicating that little was to be gained by more sophisticated (and expensive) systems in such a situation. Memoryless sources are of less interest than sources with memory since the potential gains of data compression are usually more for the latter as

they have more "redundancy" to remove. Simulations indicate that high rate predictive quantization often yield good performance for sources with memory, but there is no rigorous counterpart to the results of [4,5] (despite some prevalent myths). Proving that such systems in general (or even in particular cases) yield nearly optimal performance is one of the important open problems in information theory. In addition, little is known about what schemes work well when a low rate such as 1 bit/symbol is required (wherein PCM may be significantly suboptimal even for a memoryless source).

Recently results have been proved in information theory that show that nearly optimal performance can be achieved by a class of data compression systems called tree or trellis encoding systems [6, 7, 8, 9, 10]. Those systems consist of a (possibly nonlinear) digital filter as a decoder and a matched tree or trellis search as the encoder. Furthermore, it has been shown that the decoding filter can be assumed to be time-invariant [10] in which case the system is referred to as a sliding block code [11, 12]. Such coding structures are of potential practical importance since similar structures have been found to yield nearly optimal performance with moderate complexity in the dual problem of channel coding for error correction.

Viterbi algorithms, which are used in the trellis code case (when the decoder is a finite state device), are now fairly cheap to implement and the decoder is simply a digital filter.

We are considering systems that have the structure described in Fig. 1. The decoder consists of a shift register (finite or infinite) and a (usually nonlinear) function $F:A^K \to \mathbb{R}$ where A is the alphabet

of the decoder input sequence, K the decoder shift register length and $\mathbb{R}$ the real line. The encoder is a search algorithm matched to the decoder; a tree search or a trellis search.

As an example, a rate 1 system has a decoder that at each time unit accepts a binary symbol and outputs a reproduction symbol. Since the current output depends only on the current binary input and the previous contents of the shift register (the state of the decoder) the decoder behavior can be described by the tree diagram of Fig. 2 (assuming an initial all zero state). Each tree node represents a state and one advance on the upper path if "1" is input and on the lower path if "0" is input and outputs the "label" on the branch taken.

If the decoding register has a finite size K (there is a finite memory and no feedback) then the tree has a simplified picture obtained by realizing that there are only 2^{K-1} distinct states and hence many of the tree nodes can be "merged." The resulting structure called a (time-invariant) trellis is depicted in Figure 3 for the case K = 3.

Since the decoder uniquely determines the trellis or tree, the whole problem of system design becomes the problem of choosing the best, or at least a good, decoder. Once a decoder is given, the encoder is just a search algorithm that searches the trellis or the tree and finds a good path.

In practical systems the delay must be finite and hence the search algorithm will usually fall within one of the two categories:

(1) Block search - The encoder inputs a block of source symbols

then searches for a sequence that will yield the best reproduction when driving the decoder. When the best path is found, the encoder outputs the whole string and proceeds to the next non-overlapping block. After each block the encoder may drive the decoder back to the all zero state or may continue from the last state of the previous block. If the decoder is finite and the search algorithm is the Viterbi algorithm this system is referred to as a Block Trellis Encoding System.

(2) Incremental Search - The encoder inputs a block of source symbols and searches for the best path, then the first step along this path is outputted. The next input symbol is read in the best path is found and the first step is transmitted for the second overlapping block and so on. In executing the search, information from previous searches can be used to reduce the computation time significantly. This system is an example of a true sliding block code.

For a trellis search the Viterbi algorithm [13, 14] is used almost universally. For a tree search one can use a simple exhaustive search (if the block is very short) or the Stack, Fano or M-L algorithm [15, 1 (Chapter 6)].

As mentioned before, the design problem of a tree or trellis coding system is the problem of designing the nonlinear filter since this in turn specifies the tree or trellis search by defining the tree or trellis. The design problem is sometimes described as that of finding a rule to "color" or "label" the tree.

It was recently proved that tree and trellis codes, operating in the

block search fashion can achieve performance arbitrarily close to the theoretical rate-distortion bound [10]. Although it is believed that incremental tree or trellis search system can also achieve such performance, the proof is still an open problem.

As is often the case in information theory, the coding theorems only prove the existence of good codes having a certain structure, they do not say how to actually design such a good code. The few attempts to study the actual performance of specific tree and trellis encoding systems have fallen into the following three categories.

(1) Random Coding - The traditional proofs of the tree source coding theorems involve random coding arguments, that is, selecting the branch labels at random according to a probability distribution arising in the evaluation of the optimal achievable performance (distortion-rate or rate-distortion function) [6, 7, 8, 9]. This led to the design of tree codes by such random "coloring" of the trellis [16]. Unfortunately, however, this method results in a time-varying decoding filter and hence more complicated decoding circuitry and a vastly more complicated tree with large storage requirements.

(2) Adaptive Data-Filtering - Mark [17] developed a technique for designing a decoder by adaptive Kalman-type filtering of the source data to produce a source model with which to color the trellis. This technique outperforms predictive quantization, but it also requires a time-varying decoding filter and the transmission of extra "side information" to tell the decoder how the trellis is colored.

(3) Plagiarized Decoder - Several authors [18, 19, 20] suggested a very intuitive technique whereby tree encoding is used to improve traditional techniques; simply take the usual decoder of a data compression system (e.g., predictive quantization or delta modulation) but use a tree search algorithm as the encoder. Since an exhaustive tree search will find the best path through the tree while the predictive quantizer finds a path through the tree, the performance of the new system cannot be worse than that of the traditional system and actually a considerable gain is often realized. For a practical tree search which is not exhaustive (usually because of computation and storage requirements) the gain is somewhat smaller.

In this paper several techniques for tree and trellis encoding are studied and their performance is compared with that of the traditional techniques and the theoretical bounds. We focus on time-invariant decoders and consider one bit per symbol data compression systems both for simplicity and since this is a less well-understood case. All the results are readily extended to higher rates. To allow comparison with the traditional techniques of PCM, DPCM, DM and predictive quantization, we consider the common source models of Gaussian: memoryless, autoregressive and moving average processes. Most of the techniques presented are extendable to a much larger class of sources. The memoryless case may at first seem to be beating a dead horse but it is useful both for comparison with PCM in the low bit rate case and as a step in designing trellis

codes for sources that can be modelled as a filtered memoryless process. It is worth noting that simple one bit tree or trellis encoding systems can provide about .7 dB improvement in SNR over the optimal one bit quantizer for a memoryless Gaussian source. To the best of our knowledge such improvement has not previously been realized without a random coding approach (and hence time-varying systems) in this case. Also, this apparently low improvement is significant since the total improvement cannot be greater than 1.4 dB.

Several interesting properties of these systems and open problems in their analyses are described. We believe that their simplicity and superiority over traditional techniques make them a hopeful class of systems.

Most of the forthcoming results were obtained by simulations using the INTERDATA minicomputer running under DOS and PDP-11 minicomputer running under the UNIX system.

2. Preliminaries

For simplicity and to facilitate comparison with existing systems we here consider only first-order linear Gaussian sources, that is, sources modelled as white Gaussian sources filtered by first-order linear filters. Let $\{Z_n\}$ be a sequence of independent, identically distributed (i.i.d., memoryless, white) Gaussian zero mean random variables with variance $\sigma_Z^2 = E\, Z_n^2$. A first order Gaussian autoregressive source $\{X_n\}$ (or a Gauss Markov source) is defined by the difference equation

$$X_n = Z_n + a\, X_{n-1} \quad , \tag{2.1}$$

where a is called the autoregressive constant. In other words, X_n is the output of the first-order autoregressive filter of Fig. 4 driven by Z_n. If $|a| < 1$, then the filter is stable and $\{X_n\}$ is stationary and (2.1) makes sense for all integers n and we can write

$$X_n = \sum_{k=0}^{\infty} a^k Z_{n-1} \tag{2.2}$$

(where $a^0 = 1$) if $|a| \geq 1$. If $|a| = 1$ ($\{X_n\}$ is the discrete-time Wiener process), than the filter is unstable and (2.1) must be "initialized" to make sense, e.g., set $X_0 = 0$ and let (2.1) hold for $n \geq 1$. The resulting process is nonstationary. The process $\{X_n\}$ is easily seen to have zero mean and, if $|a| < 1$, to have variance $\sigma_X^2 = \sigma_Z^2 \sum_{k=0}^{\infty} a^{2k} = \sigma_Z^2/(1-a^2)$, correlation coefficient $r = E\, X_n X_{n-1} = E\, X_{n-1}(Z_n + a\, X_{n-1})/\sigma_X^2 = aE\, X_{n-1}^2/\sigma_X^2 = a$, and autocorrelation $R_X(\tau) = \sigma_Z^2 a^{|\tau|}$. A first-order Gaussian moving average source is defined by the difference equation

$$X_n = Z_n + a Z_{n-1} \tag{2.3}$$

and is obtained by passing $\{Z_n\}$ through the first-order moving average filter of Fig. 5. The filter is stable, the resulting process $\{X_n\}$ is stationary and has autocorrelation $R_x(k) = E X_0 X_k = \sigma_Z^2\{(1+a^2)\delta_{0,k} + a\delta_{-1,k} + a\delta_{1,k}\}$, where $\delta_{m,k} = 1$ if $m = k$ and 0 if $m \neq k$.

The quantization techniques of simple quantization (PCM), predictive quantization, delta modulation, and differential pulse code modulation (DPCM) can all be considered as special cases of the predictive quantization scheme of Fig. 6. The binary coder is a fixed rate one-to-one operation mapping each quantized error term $\hat{e}_n$ into a binary k-tuple, yielding a rate $\log_2 k$ per symbol data compressor. A PCM system has the trivial predictor $P_n = 0$. A DPCM system sets $P_n = \hat{X}_{n-1}$ and a delta modulator is simply a one-bit DPCM system and hence both systems simply quantize and transmit the error between the current sample X_n and the reconstructed previous sample $\hat{X}_{n-1}$. A linear predictive quantizer sets $P_n = a_1\hat{X}_{n-1} + a_2\hat{X}_{n-2} + \cdots$, a scaled version of the previous reconstructed samples. In general, one would like to choose an "optimum" predictor, but the complexity of the feedback loop precludes an exact determination of such a predictor. As a result, most designers use as the predictor P the optimum linear predictor of X_n given $X_{n-1}, X_{n-2}, \ldots$ the <u>actual</u> previous samples rather than their reconstructed value. The argument is that if the quantizer is "good," $\hat{X}_{n-1}$ will be nearly X_{n-1} and the optimal linear predictor for X_{n-1} should also work for $\hat{X}_{n-1}$. For example, if $\{X_n\}$ is a zero mean stationary first order Markov source, then the optimal estimate $X_n = f(X_{n-1}, X_{n-2}, \ldots)$

for X_n in the sense of minimizing the expected squared error $E(X_n-P_n)^2$ is well-known to be $f(X_{n-1},X_{n-2},\ldots) = r\,X_{n-1}$, where $r = E\,X_nX_{n-1}/E\,X_n^2 = R_X(1)/\sigma_X^2$, the correlation coefficient of the source. We shall therefore usually confine interest to a linear predictive quantizer of the form of Fig. 7. Note that for such a system the decoder is a memoryless nonlinearity (mapping binary symbols into a finite subset of real numbers) followed by a linear filter (with feedback) satisfying the relation

$$\hat{X}_n = \hat{e}_n + a\,\hat{X}_{n-1} = \ldots = \sum_{k=0}^{\infty} \hat{e}_{n-k}a^k \qquad (2.4)$$

where $a = 0$ $(0^0 = 1)$ for PCM, $a = 1$ for DPCM and delta modulation, and $a = r$ for an optimum linear predictive quantizer for a stationary zero mean first-order Markov source. If more complicated predictors are used, the filter defined by (2.1) becomes more complicated. Note that even in the simple case of linear predictive quantization, the decoder has the form $\hat{X}_n = g(Y_n, Y_{n-1}, Y_{n-2},\ldots)$, that is, the decoder has an infinite constraint length.

For the case of a Gauss Markov source, it is well-known that the optimal linear predictor yields an error sequence $\epsilon_n = X_n - r\,X_{n-1} = Z_n$ that is i.i.d., that is, "whitening" X_n by removing the optimal linear prediction yields an error process ϵ_n (called the innovations of $\{X_n\}$) that is simply the original white process $\{Z_n\}$ used to model $\{X_n\}$. As previously mentioned, it is a standard approximation for high rate predictive quantization systems for Gauss Markov sources to assume $\epsilon_n \tilde{=} e_n$, that is, the quantizer is so good that $\hat{X}_{n-1} \tilde{=} X_{n-1}$ and hence $e_n = X_n-P_n = X_n-r\,\hat{X}_{n-1} \tilde{=} X_n-r\,X_{n-1} = \epsilon_n$. This

assumption is equivalent to replacing the predictive quantizer of Fig. 7 by the innovations quantizer of Fig. 8 for purposes of analysis. An approximate analysis of predictive quantization is then obtained by using asymptotic quantizer relations [4] on ϵ_n to obtain $E(\epsilon_n - q(\epsilon_n))^2$ and using the fact $\epsilon_n \tilde{=} e_n$ to state

$$E(e_n - q(e_n))^2 = E(X_n - \hat{X}_n)^2 \tilde{=} E(\epsilon_n - q(\epsilon_n))^2 \tag{2.5}$$

where the middle equality follows from the fact that

$$\begin{aligned} E(X_n - \hat{X}_n)^2 &= E(X_n - (r\hat{X}_{n-1} + q(e_n))^2 \\ &= E(e_n - q(e_n))^2 \end{aligned} \tag{2.6}$$

The complication of the nonlinear feedback loop in the system has thus far precluded a rigorous proof of this approximation, but it has been found to be good for high rate systems. In addition, the innovations quantizer has been thought to be itself a reasonable system for high rate data compression since it is known that for Gauss Markov sources the optimal performance as given by Shannon's rate-distortion function is the same for the source and its innovations [21], suggesting a good data compressor can be built that operates on the memoryless innovations. The innovations quantizer is considered here only to try to dispel the myth by demonstrating that in general a data compressor operating on the innovations in a nearly optimal fashion can in fact perform quite poorly overall (for example, in the Gauss-Markov case the optimal innovation quantizer will perform exactly the same as an optimum quantization of the process itself) and that neither approximation to a predictive quantizer nor an interesting but not well-understood result

of rate-distortion theory can be used as motivation for using an innovations quantizer.

A tree or trellis data compressor consists of a sliding-block decoder g mapping binary sequences into reproduction symbols and a matched tree search algorithm [6,7,8,9,10]. The encoder, which knows the current state of the decoder, uses a distortion measure ρ (here $\rho(a,b) = (a-b)^2$), a search depth L and the current state of the decoder register to find the available sequence of path labels, say $\hat{x}_0,\ldots,\hat{x}_{L-1}$, emanating from the initial state (node) of the tree that minimizes the sample distortion $L^{-1}\sum_{i=0}^{L-1} \rho(x_i,\hat{x}_i)$ between reproduction sequence and the source sequence $x_0,\ldots,x_{L-1}$. If it is a block tree code [6,7,8,9,10], the encoder then outputs the binary "path map" through the tree forcing the decoder to output $\hat{x}_0,\ldots,\hat{x}_{L-1}$. If it is an incremental tree code [10,12,18], the encoder outputs only enough binary symbols (one for a rate 1 code) to force the decoder to output $\hat{x}_0$. After the binary channel symbols (L or 1) are output, a new search is performed and operation continues. The actual tree search may be performed via the Fano, Stack, or (M,L) [15, 18, 29, 30, 1 (Ch. 6)] algorithms or, if the decoder has finite constraint length and hence the tree reduces to a trellis, by a Viterbi algorithm [9,10].

A natural question when dealing with data compression systems is what the "optimal" performance is in the sense of minimizing average distortion (or "maximizing" fidelity) for a given fixed rate. Given a probabilistic model μ for a source, a distortion measure ρ, and a fixed rate R, an unbeatable lower bound to the average distortion between the original source and its reconstruction is given by Shannon's

distortion-rate function D(R) [1,2]. Roughly speaking D(R) is the smallest average distortion over all "stochastic codes" or random connections between a source and an approximation of the source such that the average mutual information rate between source and approximation is less than R. In addition, coding theorems exist showing that performance arbitrarily close to D(R) can be achieved using certain classes of coding structures, in particular using sliding-block encoder and decoder [11,12,22] and a sliding-block decoder with a matched block tree search encoder [10]. No coding theorems exist for the incremental tree encoder, but simulations have often shown these encoders to be superior to the block tree encoder. For our purposes, however, D(R) provides an absolute yardstick to which all actual systems can be compared.

3. Plagiarized Decoders

As previously described a natural choice for a decoder in a tree or trellis encoding data compression system is simply the decoder of a good existing system such as linear predictive quantizer or delta-modulation. As the decoder for such systems has an infinite constraint length, a tree search algorithm is required. Alternatively, one may truncate the decoder to a finite length K by replacing (2.1) by

$$\hat{X}_n = b \sum_{i=0}^{K-1} \hat{e}_{n-i} a^i \tag{3.1}$$

and then use the Viterbi algorithm to search the resulting trellis. The scaling constant b allows the designer some extra freedom and was selected so as to experimentally minimize $E(X_n - \hat{X}_n)^2$ for a given constraint length K. In this section we concentrate on the predictive quantizer decoder and postpone the delta modulator decoder to the next section since it can be arrived at using the fake process approach and we prefer to call this decoder the Central Limit Theorem decoder (CLT) for reasons which will be clarified later.

The truncated predictive quantizer decoder is described in Fig. 9. Simulations were performed for block-Viterbi algorithm encoders with decoders of lengths 3,4 and 5 and first order Gauss Markov sources with various autoregressive constants. The block length was 2000 symbols. The experimentally determined values of b is 0.7-0.8 and the SNR in dB

$$\text{SNR} = 10 \log_{10} \frac{E X_n^2}{E(X_n - \hat{X}_n)^2} \tag{3.2}$$

is plotted in Fig. 10. Also plotted for comparison is the performance

of linear predictive quantization with predictor

$$P_n = r \hat{X}_{n-1}$$

and an optimum quantizer as determined by Arnstein [23]. The distortion rate function is also plotted for reference. Note that this trellis code system outperforms the predictive quantizer by about 1 dB for autoregressive constants and are not too close to one. This is because truncating the decoder causes the system to be unable to take advantage of the large amount of redundancy when the autoregressive constant is large.

If the decoder is not truncated and a tree search algorithm is used instead of a trellis search, then the performance for large autoregressive constant is improved. An incremental exhaustive tree search was used in the simulation; this algorithm has the advantage of operating in a sliding block fashion thus requiring a constant computation time for each symbol processed and thus the need for buffers, is eliminated in practical systems. Some tree search algorithms, like the Stack and Fano algorithms have search times which depend on the particular source sequence. It was also found that the stack-algorithm required much more computation time without any performance improvement in our case.

Because of the constant computation time and constant delay the incremental tree search is especially attractive for practical real time systems since it was found experimentally that for sources with autoregressive constants not too close to one a relatively low search depth of the order of 5 is sufficient. Thus for each input symbol the encoder must check 32 alternatives, which is a reasonable number.

Simulation results for the predictive quantizer decoder and the incremental tree search encoder are given in Fig. 11 for two search depths. The experimentally determined value of b is 0.8-0.9 over the whole range.

It should be noted that the predictive quantizer decoder-tree search encoder system (which can be denoted by LAPQ, Look Ahead Predictive Quantization) does not suffer a deterioration in performance for high autoregressive constants as there is no decoder truncation in this case.

4. Fake Processes

The fake process approach is a design philosophy that allows one to substitute the problem of designing a tree or trellis data compression system with the problem of designing a filter that, when driven by an i.i.d. discrete source with, say, M equally probable letters, will produce a "good" fake of the process one wishes to compress.

The fake process technique is a means of mechanizing the theoretical approach developed in [10]. Originally this design problem was called the simulation problem but we prefer the name "fake process" to avoid confusion as most results are obtained by simulations.

The fake process technique is intimately connected with the notion of a distance between stochastic processes. *Several definitions* of such a distance exist but the one applicable to our case is the generalized Ornstein distance [24].

Suppose that we are given two stationary-ergodic processes $\{X\}$ and $\{Y\}$ and a single letter distortion measure $\rho(\cdot,\cdot)$. The processes are completely specified by the n^{th} order marginal distributions ν^n and μ^n for $n = 1,2,3,\ldots$ for $\{X\}$ and $\{Y\}$ respectively.

Now define the n^{th} order generalized Ornstein distance by

$$\bar{\rho}_n(X^n,Y^n) = \inf E\left\{n^{-n}\sum_{i=0}^{n-1}\rho(X_i,Y_i)\right\} \tag{4.1}$$

where the infinmum is taken over all stationary probability distributions having ν^n and μ^n as marginals.

Define the generalized Ornstein distance $\bar{\rho}$ by

$$\bar{\rho}(X,Y) = \sup_n \bar{\rho}_n \tag{4.2}$$

Roughly speaking $\bar{\rho}$ is the minimal distribution between the processes $\{X\}$ and $\{Y\}$ by embedding them both in a super process that yields $\{X\}$ and $\{Y\}$ as coordinate processes.

Since a coding process is a means of statistically joining two processes (the source and the reproduction) this notion of distance has several applications to information theory [24, 25].

Armed with this notion of distance we can now define a fake process. Suppose that we are given a stationary-ergodic process $\{X\}$ and a single letter distortion measure $\rho(\cdot,\cdot)$. Let $\{U\}$ be an i.i.d. process uniformly distributed on M discrete levels (thus the entropy rate of $\{U\}$ is $\log_2 M$). Given $\{U\}$ and $\epsilon > 0$ find a filter (such as in Fig. 12) such that when it is driven by the process $\{U\}$ the output process $\{\hat{X}\}$ satisfies

$$\bar{\rho}(\hat{X},X) = \epsilon_0 \leq \epsilon \tag{4.3}$$

$\{\hat{X}\}$ is called a fake process for $\{X\}$ and ϵ_0 is the distance between the real and fake processes.

If $D(R)$ is the distortion rate function of the process $\{X\}$ relative to the fidelity criterion $\rho(\cdot,\cdot)$ then it can be shown [10] that for any fake process $\{\hat{X}\}$

$$\bar{\rho}(\hat{X},X) \geq D(\log_2 M) \tag{4.4}$$

for any choice of filter. In addition for a large class of processes (including all Markov processes) given any $\epsilon > 0$, a fake process (or a filter) can be found such that

$$\bar{\rho}(\hat{X},X) < D(\log_2 M) + \epsilon \tag{4.5}$$

The appearance of the distortion-rate function in (4.4) and (4.5) immediately suggests that the fake process technique should be applicable to source coding (or data compression) which is indeed the case. Suppose we have a good fake process (i.e., $\bar{\rho}$ is close to $D(\log_2 M)$). Use the filter that produces the fake process as the decoder in a rate $\log_2 M$ data compression system and use a tree or Viterbi algorithm as the encoder. Since the search algorithm essentially finds the best path through the tree or trellis, the performance of this system for a sufficiently large block length should be close to the distance between $\{X\}$ and $\{\hat{X}\}$ which in turn is close to $D(\log_2 M)$, thus we have designed a good data compression system using a good fake process. The above discussion can be made precise as is done in [10] to show that good fake processes indeed yield good data compression systems.

We have shown that in order to design good data compression systems one can reduce the problem to that of designing good fake processes. In many cases a good fake process is easy to arrive at because of a special source structure or property and, as will be seen, there are some natural ways to obtain good fake processes. Unfortunately, theoretical analysis of the performance of the suggested systems is at this point impossible because of the analytical difficulty in computing the $\bar{\rho}$-distance except for simple cases.

It also might be worth noting that we have obtained theoretical bounds on the fidelity of simulating a continuous physical process by filtering a finite entropy i.i.d. process. Since, for example, a computer is a discrete machine, so all computer-produced processes used to simulate continuous processes are fake processes and the bounds (4.4), (4.5) apply.

The C.L.T. Decoder

The Central Limit Theorem (C.L.T.) states that the distribution of the sum of a large number of properly scaled finite variance i.i.d. random variables is well-approximated by a Gaussian distribution. To be precise if $\{Y_n\}$ are i.i.d. with zero mean and variance σ^2 then

$$\Pr\left(\frac{1}{\sigma\sqrt{n}} \sum_{i=1}^{n} Y_i \leq a\right) \rightarrow \int_{-\infty}^{a/\sigma} \frac{1}{\sqrt{2\pi}} e^{-t^2/2}\, dt \tag{4.6}$$

as $n \rightarrow \infty$.

The summing and scaling can be thought of as a sliding-block coding operation and this result suggests a means of generating a fake Gaussian process with an entropy rate constraint. The resulting filter (or decoder) is shown in Fig. 13. Since we have

$$\hat{X}_n = b \sum_{i=0}^{K} U_{n-i+1} \tag{4.7}$$

the filter is really a K^{th} order linear moving average filter. Also if U_n takes on the values $\pm 1, \pm 2, \pm 3, \ldots, \pm(2^M-1)$ then the decoder is really the truncated DPCM decoder. For the case $M = 1$, this is the truncated delta modulation decoder and it is interesting that one can arrive at this decoder using the fake process approach.

Note that it is not claimed that $\{\hat{X}_n\}$ is a Gaussian process, but only that it has approximately Gaussian marginals. In addition, $\hat{X}_n$ will not be independent, as a matter of fact the following interesting properties of the C.L.T. are proved in Appendix A:

(1) $H(\hat{X}) = \log_2 M = H(U)$ where M is the number of discrete levels the process $\{U_n\}$ takes on. This is independent of the shift register length.

(2) The autocorrelation function of $\{\hat{X}_n\}$ is given by

$$R_{\hat{X}}(\tau) = E\,\hat{X}_0\hat{X}_\tau = \begin{cases} \sigma^2(K-|\tau|)/K & |\tau| \le K \\ 0 & \text{elsewhere} \end{cases}$$

where K is the shift register length.

It can also be shown [26] that $\{\hat{X}_n\}$ is not a Markov of any order. Because of the strong memory dependence it seems that the C.L.T. fake process should be used to code sources with memory.

Suppose that the source is a first order Gauss-Markov process with autoregressive constant a. If U is a Bernouli process (we deal with rate 1 data compression systems) then it can be shown [26] that the scaling constant b in (3.1) should be

$$b = \frac{1}{\sqrt{K(1-a^2)}} \tag{4.8}$$

if we wish the fake process to have the variance of the original process (K is the shift register length). This value of b is in very good agreement with the experimentally found value and is given in Fig. 14 for a shift register of length 5, and various autoregressive constants.

The performance of the data compression system consisting of the C.L.T. decoder and a Viterbi algorithm encoder was evaluated through simulations using several blocks of length 2000 of a first order Gauss-Markov source. The results are given in Fig. 14 for various decoder lengths and autoregressive constants. It can be seen that the C.L.T. system outperforms delta modulation by about 2 dB in the mid-range autoregressive constants, for very high autoregressive constants the performance of the C.L.T. deteriorates due to a short shift register length.

It is possible to have an infinite length C.L.T. decoder, but in this case the trellis structure is lost and a tree search is used as the encoder. We chose to use an incremental tree search for reasons outlined in the introduction (this results in a true sliding block system; sliding block encoder and sliding block decoder). We chose to call this system LADM for Look Ahead Delta Modulation since it is basically a delta modulation system that looks into the "future" to determine the current output. The performance of the system, based upon a simulation of 10,000 input samples is given in Fig. 15. In this example a search depth of 5 was used as we found out that bigger searches contributed negligible improvement. A constant improvement of about 1.5 dB is realized (relative to delta-modulation), as a matter of fact even for a Wiener sequence (unity autoregressive constant) LADM will yield distortion (mean square error) that is 1.5 dB below the delta-modulation distortion (the same is true for the SNR defined as $10 \log_{10} \frac{\frac{1}{n}\sum_{i=0}^{n} (x_i - \hat{x}_i)^2}{\frac{1}{n}\sum_{i=0}^{n} \hat{x}_i^2}$). Note also that the complexity increase for the LADM system is moderate since for a depth 5 search the encoder has to search only 32 possibilities before deciding upon the output.

Variants of the C.L.T. Decoder

There are several ways to modify the C.L.T. decoder in such a way that the autocorrelation of the output sequence is decreased thus making these decoders candidates for compression of i.i.d. Gaussian processes.

Three examples are shown in Fig. 16*. For each decoder the autocorrelation function is also shown. Assume that the input is an i.i.d. Bernouli process with levels +1 and -1. Note that the decoder in Fig. 16(a) has the property that any different shifts have at most one common bit thus the autocorrelation is reduced significantly. The decoder in Fig. 16 (b) is a "brute force" way to obtain zero autocorrelation by using the first bit as a sign bit.

Decoders of the form of Fig. 16 were tested in a data compression system with a Viterbi algorithm as the encoder and an i.i.d. Gaussian source. The performance was quite poor. The decoder of Fig. 16 (a) and 16 (c) were able approximately to match the performance of the optimum quantizer for shift register up to 7. The decoder in Fig. 16 (b) performs about 0.2 dB worse than the optimum quantizer. Note that as far as distribution and second order characteristics (spectrum) are concerned, the decoder in Fig. 16 (b) is a better match for the i.i.d. Gaussian process which shows that even though distribution and second-order characteristics can be used as a guideline to design fake processes, a good match of those characteristics does not necessarily yield a good fake process.

*The first two were suggested by L.D. Davisson, the last one by J. Dunham.

Inverse-Distribution Decoders

Given a random variable X with a continuous cummulative distribution $F_X(x) = \Pr(X \le x)$, then the random variable X can always be modelled as $X = F_X^{-1}(V)$ where V is a uniformly distributed random variable on $[0,1]$ and $F_X^{-1}(a) = x$ if x is the smallest value for which $F_X(x) = a$. In other words, the random variable $F_X^{-1}(V)$ has a distribution identical to that of X since (when F is continuous):

$$F_{F_X^{-1}}(v) = \Pr\{F_X^{-1}(V) \le v)\} = \Pr\{V \le F_X(v)\} = F_X(v) \qquad (4.9)$$

when F is not continuous (4.9) still holds [26],

This suggest another technique for faking a Gaussian process: consider the filter register as a binary expansion of $[0,1]$ and take F_X^{-1} of the contents. We focus on the rate 1 case for simplicity. Given an i.i.d. binary $\{0,1\}$ random process $\{U_n\}$ with $\Pr(U_n=0)=\Pr(U_n=1) = 1/2$ the contents of a length K shift register say $(U_n, U_{n-1},\ldots U_{n-K+1})$ can be considered as a finite approximation $\tilde{v}_n$ to a uniform variable, that is the binary K-tuple $(U_n,U_{n-1},\ldots,U_{n-K+1})$ is considered as a binary representation of a number between 0 and 1 via

$$\tilde{v}_n^{(K)} = \sum_{i=0}^{K-1} (U_{n-i}2^{-i-1}) + 2^{-(K+1)} \qquad (4.10)$$

where the bias $2^{-(K+1)}$ is used for technical reasons to avoid getting $\tilde{v}_n^{(K)} = 0$. For a fixed K, $\tilde{v}_n^{(K)}$ is a discrete random variable uniformly distributed on the numbers $\{i2^{-K} + 2^{-(K+1)}, i=0,1,2\ldots,2^K-1\}$. In the limit as $K \to \infty$ the distribution for $F_X^{-1}(\tilde{v}_0^{(K)})$ converges to that of $F_X^{-1}(V)$ which is F_X. Thus in the limit, for large K, the sliding-block coder of Fig. 17 driven by a binary i.i.d. equiprobable sequence $\{U_n\}$ should produce a process

$$\hat{X}_n = F_X^{-1}(\tilde{v}_n^{(K)}) = F_X^{-1}\left(\sum_{i=1}^{K} U_{n-i-1}2^{-i} + 2^{-(K+1)}\right) \qquad (4.11)$$

with a distribution which is approximately F_X and which has the following properties [26] which are proved in Appendix B:

(1) $\{\hat{X}_n\}$ is a first-order Markov process.

(2) The entropy of $\hat{X}$ is given by

$$H(\hat{X}) = H(U) = 1$$

for a driving binary equiprobable source.

(3) The process $\{\hat{X}_n\}$ has the distribution of an equal area (maximum entropy) quantization of the distribution F_X where the output levels are chosen so as to minimize the average absolute error.

(4) The autocorrelation function $R_{\hat{X}}(\tau)$ satisfies $R_{\hat{X}}(\tau) = 0$ as $\tau \geq K$ for decoder length K. For $K = \infty$, $R_{\hat{X}}(\tau) \to [E(x)]^2$ as $\tau \to \infty$; however, the rate of convergence depends upon the distribution itself.

Obviously, successive samples of the fake process cannot be independent since each successive binary K-tuple is a shift of the previous with one new symbol going in and the last symbol being shifted out. Here, however, a simple technique can be used to at least cause the $\hat{X}_n$'s to be uncorrelated (or "linearly independent") and hence $\{\hat{X}_n\}$ will be a pseudo-white process in the sense of having a flat power spectrum and approximately the right marginal distribution. This rather odd process will be a candidate for coloring the trellis for memoryless sources and will serve as a building block to color the tree for sources with memory.

The basic idea for modifying the system of Fig. 17 is to introduce

a "scrambler" in order to mix up the influence of the binary symbols in the shift register, hopefully so as to decorrelate the resulting process. Two forms of scrambling are possible: using a block scrambler $\alpha:\{0,1\}^K \rightarrow \{0,1\}^\ell$ on the binary K-tuple as in Fig. 18 or to form a scrambling function $g:[0,1] \rightarrow [0,1]$ which scrambles the discrete uniform variable $\tilde{v}_n^{(K)}$ prior to passage through the inverse distribution function as in Fig. 19.

The class of all block scramblers of Fig. 18 is contained in the class of all scrambling function scramblers as far as the input to the inverse distribution operator is concerned [26], hence we focus on the second class. The existence of decorrelating scramblers is guaranteed by the following theorem which is proved in Appendix C.

Theorem 4.1:

If the X_n's are random variables with a continuous distribution function $F_X(x)$ that is anti-symmetric about $X = \frac{1}{2}$ (i.e., the probability density function $f_X(x) = dF_X(x)/dx$ is symmetric about the mean EX, zero in our case), and if a simulation coder (fake process) is constructed as in Fig. 19 with g any function satisfying

$$g(x) + g(x + \tfrac{1}{2}) = 1 \qquad 0 \leq x < \tfrac{1}{2} \tag{4.12}$$

Then the resulting process is uncorrelated, that is:

$$E\,\hat{X}_n \hat{X}_{n+K} = 0 \quad \text{for all} \quad K \neq 0 \tag{4.13}$$

We note that the assumptions on $F_X(x)$ are satisfied for all of the Gaussian processes considered and also for other sources of interest. Also, additional constraints must be imposed upon the scrambling function

g (in addition to (4.12)) if we want the output of the scrambler to have uniform discrete distribution and also to have the maximum entropy, i.e., 1 bit/symbol. Let $[a]$ denote the integer part of a, let K be the decoder length and let A be the set

$$A = \left\{ \frac{i}{2^K} + \frac{1}{2^{K+1}} , \; i=0,1,\ldots 2^{K-1} \right\}$$

Let $g:A \rightarrow R$ then we have the following lemmas, which are proved in Appendix D.

Lemma 4.1:

The fake process of Fig. 19 will have an entropy of 1 bit/symbol if the scrambling function satisfies

$$g(x + 1/2) \neq g(x) \tag{4.14}$$

for $x \in A$ and $x < 1/2$.

Also if we let $||S||$ denote the cardinality (number of elements) in the set S and then define $g^{-1}\{X\} = \{y \in A : g(y) = X\}$ we obtain the following lemma.

Lemma 4.2:

The output of the scrambler will be a uniform discrete distribution if and only if

$$||g^{-1}\{X\}|| = ||g^{-1}\{Y\}|| \quad \text{for all} \quad X,Y \in g(A) \tag{4.15}$$

g(A) is the range of the function g.

Some examples of scrambling functions are given in Fig. 20. The reader can easily convince himself that all the functions in Fig. 20 satisfy the conditions in Lemmas 1 and 2. We concentrate on scrambling functions of the type b and c. Note that those functions are periodic on $[0,1]$ and that these functions will satisfy the conditions (4.14) and (4.15) for any number of periods. It was found out experimentally that for a shift register of length K the best results were obtained when the number of periods of the scrambling function was $2^{K-1}+1$ or $2^{K-1}-1$. In this case the output of the scrambling function of type b contains only 2^{K-1} letters (because of symmetry) while the output of a type c scrambling function contains 2^{K} letters.

Since the fake process obtained by the filter in Fig. 19 is pseudo-white we tried to use this decoder in a trellis block code system for the hard case of compressing white Gaussian noise.

Using simulations it was discovered that the scrambling function of types b and c yield systems with almost identical performance (the small deviations can be attributed to errors resulting from simulation). Thus we concentrate on the somewhat simpler class of functions of type b for the rest of this report.

i.i.d. Processes

The filter in Fig. 19 using the scrambling function of Fig. 20 (b) was used as a decoder in a data compression system where a Viterbi algorithm was used as the encoder. As usual a scaling factor was added to the output of the filter to allow for an additional degree of freedom. This scaling factor was optimized experimentally and it was found that for an i.i.d. Gaussian source, this factor was in the vicinity of .9 for all register lengths which were tried.

Simulations were carried out for several blocks of length 2000. The results are given in Fig. 21 where a comparison is made to the optimal Max quantizer [27] and the rate distortion bound. For short shift registers the improvement is very fast; however, for longer shift registers the improvement is very slow and it was impossible (because of memory and computing time limitations) to discover whether further improvement is possible for even longer shift registers. We see, however, that the improvement over the optimal quantizer is about 0.7 dB and that the resulting performance is only 0.9 dB below the rate distortion bound. As far as we know this is the only non-random source coding scheme that has been shown to beat the optimal quantizer for this source at a rate of 1 bit/symbol.

It is interesting to note that if we use the "forward channel" approach [1, pg. 101] to evaluate the "optimal" distribution at the output of this channel this turns out to be Gaussian with standard deviation of 0.87. Our decoder has an asymptotically Gaussian distribution with a standard deviation of about .9. This, rather close, agreement suggests that one should try to fake the process that arises in the calculation of the rate distortion function and not the input process itself.

Autoregressive Sources

An autoregressive source was defined by equation (2.1). Note that the process X is also the output of a linear filter with the transfer function

$$H(z) = \frac{1}{1-az^{-1}} \tag{4.16}$$

when driven by an i.i.d. process.

This leads to the possibility of faking an autoregressive process by passing the output of the filter in Fig. 19 through the linear filter of (4.16) (see Fig. 22(a)). It can be noted that this is somewhat reminiscent of the innovations quantization approach which is extremely inefficient for compressing autoregressive sources (for Gaussian sources the performance of such a system is the same as simply optimally quantizing the process X) but because of the tree search at the encoder, our system is not an innovations quantizer.

Another fake of an autoregressive source can be obtained by noting that since this is a Markov process, we can use the conditional inverse distribution $F_{X_n|X_{n-1}}(\cdot|\hat{x}_{n-1})$ instead of the marginal in Figure 19. Of course, the conditioning is made on the reproduced past $\hat{x}_{n-1}$. The filter resulting from this approach is given in Fig. 22 (b).

The filters in Fig. 22 (a) and (b) turn out to be completely equivalent [26], i.e., if they are driven by the same input sequence they will yield the same output sequence.

Another interesting result concerning the filters in Figs. 22 is that if the driving sequence $\{U\}$ is i.i.d. and uniform (on its discrete alphabet), then the output will have an exponential auto-

correlation function (this is a direct application of the fact that the output of the scrambling function is a pseudo-white noise). Thus by definition the output is a wide-sense Markov process [28].

Note that since the output of the scrambling function is only uncorrelated (and not independent), the marginal distribution on $\{\hat{X}_n\}$ is not the same as the marginal of $\{X_n\}$ even when $K \to \infty$ (K is the shift register length). A comparison between these two distributions is made in Fig. 23 for the case where $F_X(\cdot)$ is the Gaussian distribution, the decoder has a length 8 shift register and the autoregressive constant is 0.8 (results were obtained by simulating a sequence of length 100,000).

Even when the shift register of the filter in Fig. 22 is of finite length, the whole filter is not a finite state machine because of the feedback loop at the output. As a result when this filter is used as a decoder in a data-compression system, one has to use a tree search as the encoder (it is possible to use truncation also in this case and use a Viterbi Algorithm encoder; however, as we have seen this results in poor performance in the higher range of autoregressive constants).

An incremental tree search was used in simulating data compression systems using the filter in Figs. 22 as the encoder. Tests were performed on several blocks of length 2000 and the given results are the average of five tests. Results are given in Figure 24 for search depths of 5 and 8. In both cases the system outperforms the predictive quantizer, but the gain is only about 0.3 dB for the depth 5 search while it is approximately 1 dB for the depth 8 search. In

additional simulations it was shown that even the depth 5 search will yield about 1.5 dB gain (in distortion) as compared to the predictive quantizer in the case of a Wiener sequence (a = 1).

As in the i.i.d. case, the optimal scaling factor was found to be in the vicinity of .9.

Moving Average Sources

A first-order average source satisfies equation (2.3). We use the filter in Fig. 25 to obtain a fake moving average process. In this case the filter is a finite state machine for a finite shift register length and therefore it is possible to use a Viterbi algorithm as the encoder (the Viterbi algorithm is usually preferred when applicable because it is optimal).

Simulation results are given in Fig. 26 for a first-order Gaussian moving average source and a length 6 decoder. The given results are the average of five tests of length 2000 each. The rate distortion function was calculated numerically using the parametric algorithm outlined in [1, p. 115]. Results are also given for delta-modulation, but as can be expected the performance is even poorer than direct quantization. Our new system outperforms the optimal quantizer by as much as 2 dB. This is a considerable improvement for such a low rate and is obtained in a case where the common data compression schemes perform poorly.

Summary and Conclusions

The main purpose of this report was to motivate and describe the design of data compression system using the fake process approach. Results were presented for a system designed to compress various Gaussian processes as those processes are widely used to model real life phenomena. The fake process approach is not, however, limited to this type of source (with the exception of the C.L.T. decoder which uses characteristics unique to the Gaussian process). The results that were presented for low rate systems are very encouraging and the added complexity is very moderate. The next logical step would be to obtain results for higher rate systems, but very high rate systems are not practical as the complexity of trellis and tree searches grows exponentially fast with the bit rate. Also we would like to obtain results for "real life" compression schemes, such as image or speech compression using the fake process approach.

For example, to build a speech compression system "all" that is needed is a (possibly nonlinear) filter which when driven by a discrete i.i.d. source will produce a process that "sounds like" speech. There is a twofold problem with this approach; first an appropriate distortion measure ρ must be established such that the $\bar{\rho}$-distance is a good measure for the "closeness" of different sounds; and second a filter must be designed such that its output, when driven by a discrete i.i.d. source, is close in the $\bar{\rho}$ sense to human speech. If this approach can be realized, it can become an alternative to the current LPC speech compression systems which also require a considerable computational effort.

Acknowledgements

The authors wish to thank Dr. Martin E. Hellman for suggestions that he made concerning the use of scramblers and also Dr. Lee D. Davison and James Dunham for suggesting some C.L.T. derivates.

Appendix A

Properties of C.L.T. Decoder

(1) Obviously the value of the scaling factor b does not effect the entropy (as long as $b \neq 0$).

Let $S_n = (u_n, u_{n-1}, \cdots, u_{n-L+1})$ denote the state of the shift register (i.e., the values of the shift register cells). Note that if S_n is given, then S_{n+1} can take on M distinct values depending on the next value u_{n+1}. Since $\hat{X}_n$ depends only on S_n and given S_n, $\hat{X}_{n+1}$ can take on M distinct equally probable values we have

$$H(\hat{X}_n / S_{n-1}) = \log_2 M = H(U) \quad .$$

Thus

$$H(X) = \lim_{n \to \infty} H(\hat{X}_n / \hat{X}_{n-1}, \hat{X}_{n-2} \cdots) \geq \lim_{n \to \infty} H(\hat{X}_n / \hat{X}_{n-1}, \hat{X}_{n-2} \cdots S_{n-1}, S_{n-2} \cdots)$$

$$= \lim_{n \to \infty} H(\hat{X}_n / S_{n-1}, S_{n-2} \cdots)$$

$$= \lim_{n \to \infty} H(\hat{X}_n / S_{n-1}) = H(U) \quad .$$

So we have $H(X) \geq H(U)$. Since we have the reverse inequality, Property (1) is proved.

(2) Since the u_n's inside the shift register are zero mean and independent the variance of the sum is the scaled sum of variances, so:

$$\sigma^2_{\hat{X}} = R_{\hat{X}}(0) = b^2 \sigma_u^2 \cdot K \quad .$$

Also

$$\hat{X}_n = b \sum_{i=0}^{K-1} u_{n-i}$$

$$\hat{X}_{n+t} = b \sum_{i=0}^{K-1} u_{n+t-i}$$

assume $t > 0$ (we obtain the autocorrelation function for $t < 0$ by using symmetry)

$$R_{\hat{X}}(t) = E\,\hat{X}_n\hat{X}_{n+t} = b^2E\Big(\sum_{i=0}^{K-1} u_{n-i} \sum_{i=0}^{K-1} u_{n+1-i}\Big)$$

Since the u_i's are i.i.d. and zero mean all the terms in the product vanish except those of the form u_i^2. If $t \geq K$ there are no such terms and $R_{\hat{X}}(t) = 0$. If $t < K$:

$$b^2E\Big(\sum_{i=0}^{K-1} u_{n-i} \sum_{i=0}^{K-1} u_{n+t-i}\Big) = b^2E\Big(\sum_{i=0}^{K-1} u_{n-i} \sum_{i=0}^{K-1} u_{n+t-i}\Big)$$

$$= b^2E\Big(\sum_{i=0}^{K-1} u_{n-i} \sum_{i=0}^{J-t-1} u_{n-i}\Big)$$

$$= b^2(K-t)\sigma_u^2 = b^2\sigma_u^2K\,\frac{K-t}{K}$$

and this is the desired result.

Appendix B

Properties of Inverse Distribution Decoders

(1) Let $V_n = \sum_{i=1}^{K} 2^{-i} u_{n-i+1}$. By definition,

$$\hat{X}_n = F_X^{-1}(V_n + 2^{-(K+1)}) \quad . \tag{B.1}$$

Define

$$\langle V_n/2 \rangle_K = \begin{cases} \dfrac{V_n}{2} & \text{if } 2^K V_n \text{ is even} \\ \dfrac{V_n - \frac{1}{2^K}}{2} & \text{if } 2^K V_n \text{ is odd} \end{cases} \tag{B.2}$$

If we have a length K shift register containing $(u_n, u_{n-1}, \cdots, u_{n-K+1})$ the $V_n \leq 1$ is a number whose binary expansion (after the binary point) is given by $(u_n, u_{n-1}, \cdots, u_{n-K+1})$ and $\langle V_n/2 \rangle_K$ is a number whose binary expansion is given by shifting a zero into the shift register. Note that there is a one-to-one correspondence between V_n and $(u_n, u_{n-1}, \cdots, u_{n-K+1})$. Using the definition (B-2) we obtain

$$V_n = \langle V_{n-1}/2 \rangle_K + \frac{u_n}{2}$$

Since $\{u_n\}$ is i.i.d. it follows that $\{V_n\}$ is a first order Markov chain. The function $F_X^{-1}(\cdot)$ is one-to-one, thus $\hat{X}_n$ which is obtained by (B.1) is also first order Markov since

$$\hat{X}_n = F_X^{-1}(V_n + 2^{-(K+1)}) = F_X^{-1}(\langle V_{n-1}/2 \rangle_K + \frac{u_n}{2} + 2^{-(K+1)})$$

$$\hat{X}_n = F_X^{-1}\left(\left\langle \frac{F_X(\hat{X}_{n-1}) - 2^{-(K+1)}}{2} \right\rangle_K + \frac{u_n}{2} + 2^{-(K+1)} \right) \tag{B.3}$$

So $\hat{X}_n$ depends only on $\hat{X}_{n-1}$ and the next input u_n to the filter. The conclusion remains valid as $L \to \infty$.

(2) Since $\hat{X}_n$ is first order Markov then [23, Section 6.4],

$$H(\hat{X}) = \sum_{\hat{x}_n} H(\hat{X}_n/\hat{x}_{n-1}) \cdot P\{\hat{X}_{n-1} = \hat{x}_{n-1}\} \qquad (B.4)$$

If $\hat{X}_{n-1}$ is given, $\hat{X}_n$ can take on only two distinct values

$$\hat{X}_n = \begin{cases} F_X^{-1}\left(\left\langle \frac{F_X(\hat{X}_{n-1}) - 2^{-(K+1)}}{2} \right\rangle_K + 2^{-(K+1)}\right) & \text{if } u_n = 0 \\ F_X^{-1}\left(\left\langle \frac{F_X(\hat{X}_{n-1}) - 2^{-(K+1)}}{2} \right\rangle_K + \frac{1}{2} + 2^{-(K+1)}\right) & \text{if } u_n = 1 \end{cases}$$

The values of $\hat{X}_n$ are distinct since $F_X^{-1}(\cdot)$ is one-to-one. The probability distribution on those values will be the same (given $\hat{X}_{n-1}$) as the probability distribution of u_n on $\{0,1\}$ so,

$$H(\hat{X}_n/\hat{x}_{n-1}) = H(U_n) = H(U) \qquad \text{as } \{U_n\} \text{ is i.i.d.}$$

Since $H(\hat{X}_n/\hat{x}_{n-1})$ is independent of $\hat{x}_{n-1}$ using (5.15) we have

$$H(\hat{X}) = H(U) \sum_{\hat{X}_{n-1}} p(\hat{X}_{n-1} = \hat{x}_{n-1}) = H(U) .$$

Since we always assume $p\{u_n = 0\} = p\{u_n = 1\} = 1/2$, we have

$$H(\hat{X}) = 1 \text{ bit/symbol.}$$

The conclusion remains valid as $L \to \infty$.

Note that in the case where $K = \infty$, $V_n = \sum_{i=0}^{\infty} 2^{-i} u_{n-i+1}$ is a continuous random variable uniformly distributed on $[0,1]$. Since V_n is continuous then $\hat{X}_n = F_X^{-1}(V_n)$ is continuous, thus we have an example of a continuous amplitude process with an arbitrary distribution having a finite entropy.

(3) Let $S^K = \{ \frac{i}{2^K} + \frac{1}{2^{K+1}} , i=0,1,\ldots,2^{K-1}\}$. First note that if α is uniform on S^K then $Pr(\alpha = \alpha_0) = 2^{-K}$ (since there are 2^K members in S^K). Since F_X^{-1} is monotonic and continuous it is also one-to-one hence $F_X^{-1}(\cdot)$ is one-to-one. Thus $F_X^{-1}(\alpha)$ takes on 2^K distinct values with equal probability. If we choose any K bit equal area quantizer of X then it also will have 2^K distinct outputs with equal probability.

The decision level for the equal area quantizer are determined in the following way. Assume the decision levels are $-\infty = a_0, a_1, a_2, \cdots, a_N = \infty$ where for a K-bit quantizer $N = 2^K$ and we have

$$F_X(a_i) = \frac{K}{2^K} \qquad K = 0,1,2,\ldots,2^K \tag{B.5}$$

(This is the equal area condition.)

Suppose that the quantizer outputs are $\ell_0, \ell_1, \ldots, \ell_{N-1}$ where ℓ_i is output if the input is in the interval (a_i, a_{i+1}). The expected absolute error is given by

$$e = \sum_{i=1}^{N} \int_{a_i}^{a_{i+1}} |t-\ell_i| \, dF_X(t) \tag{B.6}$$

To minimize (B.5) we have to minimize each term. Rewriting the i^{th} term in the sum and taking derivatives with respect to ℓ_i the yields

$$\frac{\partial \bar{e}}{\partial \ell_i} = -\int_{a_i}^{\ell_i} dF_X(t) + \int_{\ell_i}^{a_{i+1}} dF_X(t) = 0$$

so

$$F_X(a_i) - 2F_X(\ell_i) - F_X(a_{i+1}) = 0 \quad .$$

Using (B.5) we obtain

$$F_X(\ell_i) = \frac{F_X(a_i) + F_X(a_{i+1})}{2} = \frac{i}{2^K} + \frac{1}{2^{K+1}}$$

thus

$$\ell_i = F_X^{-1}\left(\frac{L}{2^K} + \frac{1}{2^{K+1}}\right) \qquad \text{(B.7)}$$

which is the desired relation. In order to complete the proof we take second order derivative

$$\frac{\partial^2 \bar{e}_i}{\partial \ell_i^2} = -2F_X(\ell_i) < 0 \quad .$$

So indeed the ℓ_i that we found by (B.7) are the ones that minimize the absolute error.

(4) As before let

$$V_n = \sum_{i=1}^{\infty} 2^{-i} u_{n-i+1}$$

and

$$\hat{X}_n = F_X^{-1}(V_n) \quad .$$

By definition (assume $\tau > 0$)

$$R_{\hat{X}}(\tau) = E\,\hat{X}_n \hat{X}_{n+t} = E\,F_X^{-1}(V_n)F_X^{-1}(V_{n+\tau}) \quad . \qquad \text{(B.8)}$$

V_n is uniformly distributed on $[0,1]$. Also

$$V_{n+\tau} = \frac{V_n}{2^\tau} + \frac{j}{2^\tau}$$

where j is uniformly distributed on $0,1,\ldots,2^\tau-1$. (This follows from the fact that u_n is i.i.d. and uniform on $\{0,1\}$.) Since j depends only on $u_{n+\tau}, u_{n+\tau-1}, \cdots, u_{n+1}$, j is independent of V_n. Thus (B.8) becomes

$$R_{\hat{X}}(\tau) = E\, F_X^{-1}(V_n) F_X^{-1}\left(\frac{V_n}{2^\tau} + \frac{j}{2^\tau}\right)$$

$$= E\{F_X^{-1}(V_n) E(F_X^{-1}(\frac{V_n}{2^\tau} + \frac{j}{2^\tau} | V_n)\}$$

$$= \int_0^1 F_X^{-1}(t)\, \frac{1}{2^\tau} \sum_{j=0}^{2^\tau - 1} F^{-1}\left(\frac{t}{2^\tau} + \frac{j}{2^\tau}\right) dt \qquad (B.9)$$

Assume, for the moment, that $\int_0^1 F_X^{-1}(t)dt$ exists and is finite (we prove this later on); then since $F_X^{-1}(\cdot)$ is monotonic non-decreasing

$$\frac{1}{2^\tau} \sum_{j=0}^{2^\tau - 1} F_X^{-1}\left(\frac{t}{2^\tau} + \frac{j}{2^\tau}\right) \le \frac{1}{2^\tau} \sum_{j=0}^{2^\tau - 1} F_X^{-1}\left(\frac{t}{2^\tau}\right) = \frac{1}{2^\tau} \sum_{j=1}^{2^\tau} F_X^{-1}\left(\frac{j}{2^\tau}\right) \to \int_0^1 F_X^{-1}(z)dz \qquad (B.10)$$

as $K \to \infty$. This is by the Riemann sum approximation to the integral also for $t > 0$ (let $t \ge \epsilon > 0$)

$$\frac{1}{2^\tau} \sum_{j=0}^{2^\tau - 1} F_X^{-1}\left(\frac{t}{2^\tau} + \frac{j}{2^\tau}\right) \ge \frac{1}{2^\tau} \sum_{j=0}^{2^\tau - 1} F_X^{-1}\left(\frac{j+\epsilon}{2^\tau}\right) \to \int_\epsilon^1 F_X^{-1}(z)dz \qquad (B.11)$$

$$\to \int_0^1 F_X^{-1}(z)dz \quad \text{as} \quad \epsilon \to 0 .$$

Equations (B.10) and (B.11) show that

$$\lim_{\tau \to \infty} \frac{1}{2^\tau} \sum_{j=0}^{2^\tau - 1} F^{-1}\left(\frac{t}{2^\tau} + \frac{j}{2^\tau}\right) = \int_0^1 F_X^{-1}(z)dz \qquad \text{all } t \in [0,1]$$

By assumption $= \int_0^1 F^{-1}(z)dz$ is finite. Hence by using dominated convergence theorem:

$$\lim_{\tau \to \infty} R_{\hat{X}}(\tau) = \int_0^1 F_X^{-1}(t)\left(\lim_{\tau \to \infty} \frac{1}{2^\tau} \sum_{j=0}^{2^\tau - 1} F^{-1}\left(\frac{t}{2^\tau} + \frac{j}{2^\tau}\right)\right)dt$$

$$= \int_0^1 F_X^{-1}(t)dt \int_0^1 F_X^{-1}(z)dz = \left[\int_0^1 F_X^{-1}(t)dt\right]^2$$

Setting $t = F_X(z)$ yields

$$\int_0^1 F_X^{-1}(t)dt = \int_{-\infty}^{\infty} z\, dF_X(z) = E\, X$$

So $\int_0^1 F_X^{-1}(t)dt < \infty$ since $E X < \infty$. Also $E X = E \hat{X}$ since $\hat{X}_n$ has the distribution F_X.

Appendix C

Proof of Theorem 4.1:

Let K denote the shift register length. The process $\{X_n\}$ is stationary since it is a sliding block coding of an i.i.d. process. Thus by definition the autocorrelation function is given by:

$$R_X(i) = E\,\hat{X}(0)\hat{X}(i) \tag{C.1}$$

Let S_n denote the state of the shift register given by

$$S_n = (U_n, U_{n-1}, \ldots, U_{n-K+1}) \tag{C.2}$$

equivalently the state can be specified by a single integer J_n given by

$$J_n = \sum_{m=1}^{K} U_{n-m+1}\, 2^{K-m} \tag{C.3}$$

First consider the case $i < K$.

Let $[a]$ denote the integer part of a, then we have the following relation between any two states:

$$J_{n+j} = [J_n/2^j] + \sum_{m=1}^{j} U_{n+j-m+1}\, 2^{K-m} \tag{C.4}$$

for $1 \le j < K$.

Using the fact that $\{U_n\}$ is an i.i.d. Bernouli sequence with $p(U_n=0) = p(U_n=1) = \frac{1}{2}$ we can rewrite (A.1) in terms of the states $\{J_n\}$.

$$\begin{aligned} R_X(i) &= E\left\{F_X^{-1}\left\{g\left(\frac{J_0}{2^K} + \frac{1}{2^{K+1}}\right)\right\}F_X^{-1}\left\{g\left(\frac{J_i}{2^K} + \frac{1}{2^{K+1}}\right)\right\}\right\} \\ &= \frac{1}{2^{K+i}} \sum_{P=0}^{2^K-1} \left\{F_X^{-1}\left(g\left(\frac{P}{2^K} + \frac{1}{2^{K+1}}\right)\right) \sum_{m=0}^{2^i-1} F_X^{-1}\left(g\left(\frac{\left[\frac{P}{2^i}\right]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}}\right)\right)\right\} \end{aligned} \tag{C.5}$$

To show that $R_{\hat{X}}(i) = 0$ it will suffice to show that

$$\sum_{m=0}^{2^i-1} F_X^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}}\right)\right) = 0 \qquad (C.6)$$

Break the sum in (C.6) into two sums:

$$\sum_{m=0}^{2^i-1} F_X^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}}\right)\right) = \sum_{m=0}^{2^{i-1}-1} F_X^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}}\right)\right)$$

$$+ \sum_{m=g^{i-1}}^{2^i-1} F_X^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}}\right)\right) \quad . \qquad (C.7)$$

Changing index in the second sum in (C.7) we obtain:

$$\sum_{m=2^{i-1}}^{2^i-1} F_X^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}}\right)\right) = \sum_{m=0}^{2^i-1} F_X^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2} + \frac{1}{2^{K+1}}\right)\right)$$

(C.8)

and substituting (C.8) into (C.7) yields:

$$\sum_{m=0}^{2^i-1} F_X^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}}\right)\right) = \sum_{m=0}^{2^{i-1}-1} \left\{F_X^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2} + \frac{1}{2^{K+1}}\right)\right)\right.$$

$$\left. + F_X^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{1}{2^{K+1}} + \frac{1}{2}\right)\right)\right\} \quad . \qquad (C.9)$$

But we assumed $p \le 2^K - 1$ *and also we have* $m \le 2^{i-1} - 1$. Thus

$$\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}} \le \frac{1}{2} - \frac{1}{2^{K+1}} < \frac{1}{2} \qquad (C.10)$$

Thus, using the assumed property of the scrambling function we have:

$$g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}} + \frac{1}{2}\right) = 1 - g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}}\right) \qquad (C.11)$$

But the distribution F_X was assumed to be continuous and anti-symmetric around $X = \frac{1}{2}$. This implies

$$F_X^{-1}(t) = -F_X^{-1}(1-t) \qquad 0 \le t \le 1 \tag{C.12}$$

Thus using (C.11) we have:

$$F^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}} + \frac{1}{2}\right)\right) = -F^{-1}\left(g\left(\frac{[P/2^i]}{2^K} + \frac{m}{2^i} + \frac{1}{2^{K+1}}\right)\right) \tag{C.13}$$

Substituting (C.13) into (C.5) we see that all the terms in the sum are zero implying the sum in (C.6) is zero.

In the case when $i \ge K$ the states S_0 and S_i are independent which implies that $\hat{X}_0$ and $\hat{X}_i$ are independent random variables thus

$$R_X(i) = (E\,\hat{X}(0))^2 \qquad i \ge K \tag{C.14}$$

but

$$E\,\hat{X}(0) = \frac{1}{2^K} \sum_{m=0}^{2^K-1} F_X^{-1}\left(g\left(\frac{m}{2^K} + \frac{1}{2^{K+1}}\right)\right) \tag{C.15}$$

The sum in (C.15) can be shown to be zero by the same procedure that the sum in (C.6) was shown to be zero.

Appendix D

Proof of Lemmas 4.1 and 4.2

Lemma 4.1

We always have

$$H(\hat{X}) \leq H(U) = 1$$

Define J_n as in (C.3) since $\{\hat{X}\}$ is stationary:

$$H(\hat{X}) \geq H(\hat{X}_n / J_{n-1})$$

Once J_{n-1} is given $\hat{X}_n$ can take on at most two distinct values as

$$\hat{X}_n = F_X^{-1}\left(g\left(\left[\frac{J_{n-1}}{2}\right]/2^K + \frac{u_n}{2} + \frac{1}{2^{K+1}}\right)\right)$$

Since F_X^{-1} is one-to-one, X_n will take on two distinct values with equal probability for any given J_{n-1} if (4.14) holds. So $H(\hat{X}_n/J_{n-1}) = 1$ if (4.14) holds. QED.

Note that since (4.12) requires

$$g(\alpha + 1/2) + g(\alpha) = 1 \quad , \quad \alpha \leq 1/2$$

in order to satisfy (4.14) we only have to require $g(\alpha) \neq 1/2$ for

$$\alpha \in \left\{ \frac{i}{2^K} + \frac{1}{2^{K+1}} \; ; \; i=0,1,2,\ldots,2^{K-1}-1 \right\}$$

Lemma 4.2

The proof is straightforward. The input to the scrambling function is uniformly distributed on A. Suppose (4.15) holds then let $x,y \in g(A)$, with Z denoting the output and V the input to the scrambling function. Then

$$P_r\{Z=x\} = P_r\{g(V)=x\} = P_r\{V \in g^{-1}(x)\} = \frac{\|g^{-1}\{x\}\|}{2^L}$$

$$= \frac{\|g^{-1}(y)\|}{2^L} = P_r\{V \in g^{-1}(y)\} = P_r\{Z=y\} \qquad \forall x,y \in g(\mathbf{A}) \quad .$$

On the other hand assume that the output is uniformly distributed on the finite set $g(\mathbf{A})$. This implies for any $x,y \in g(\mathbf{A})$, $P_r\{Z=x\} = P_r\{Z=y\}$, so $P_r\{V \in g^{-1}(y)\} = P_r\{V \in g^{-1}(x)\}$. Since V is uniform on a discrete set this implies $\|g^{-1}(x)\| = \|g^{-1}(y)\|$. QED.

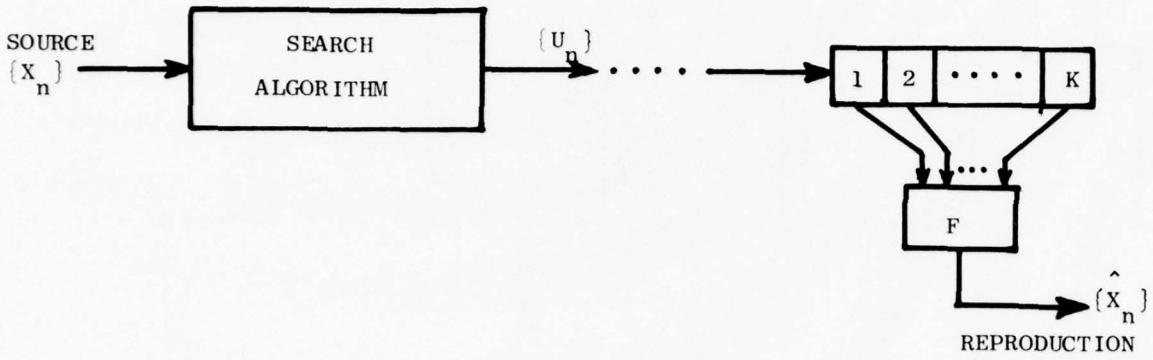


Fig. 1 THE DATA COMPRESSION SYSTEM

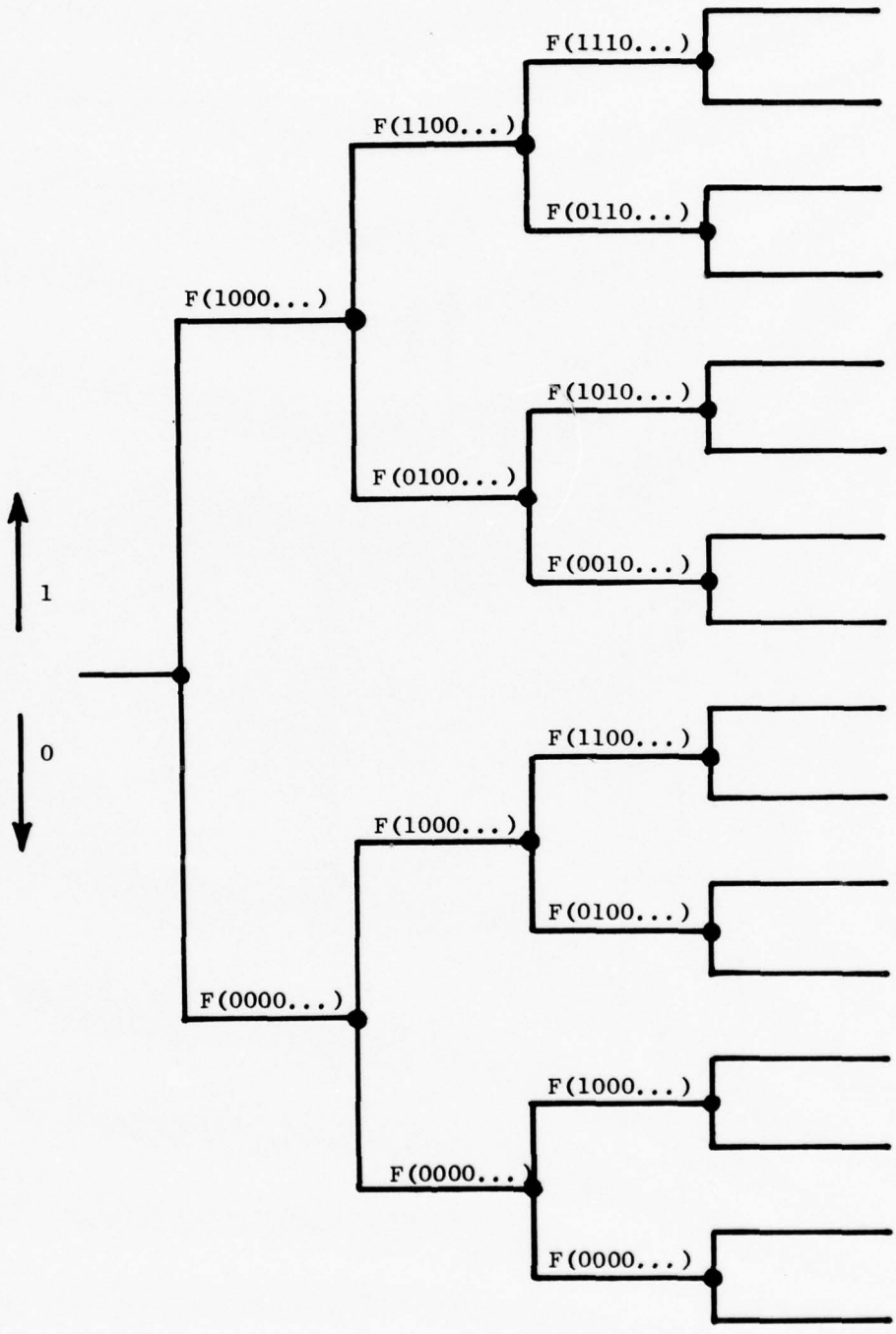


Fig. 2 A TREE DIAGRAM FOR THE DECODER IN Fig. 1

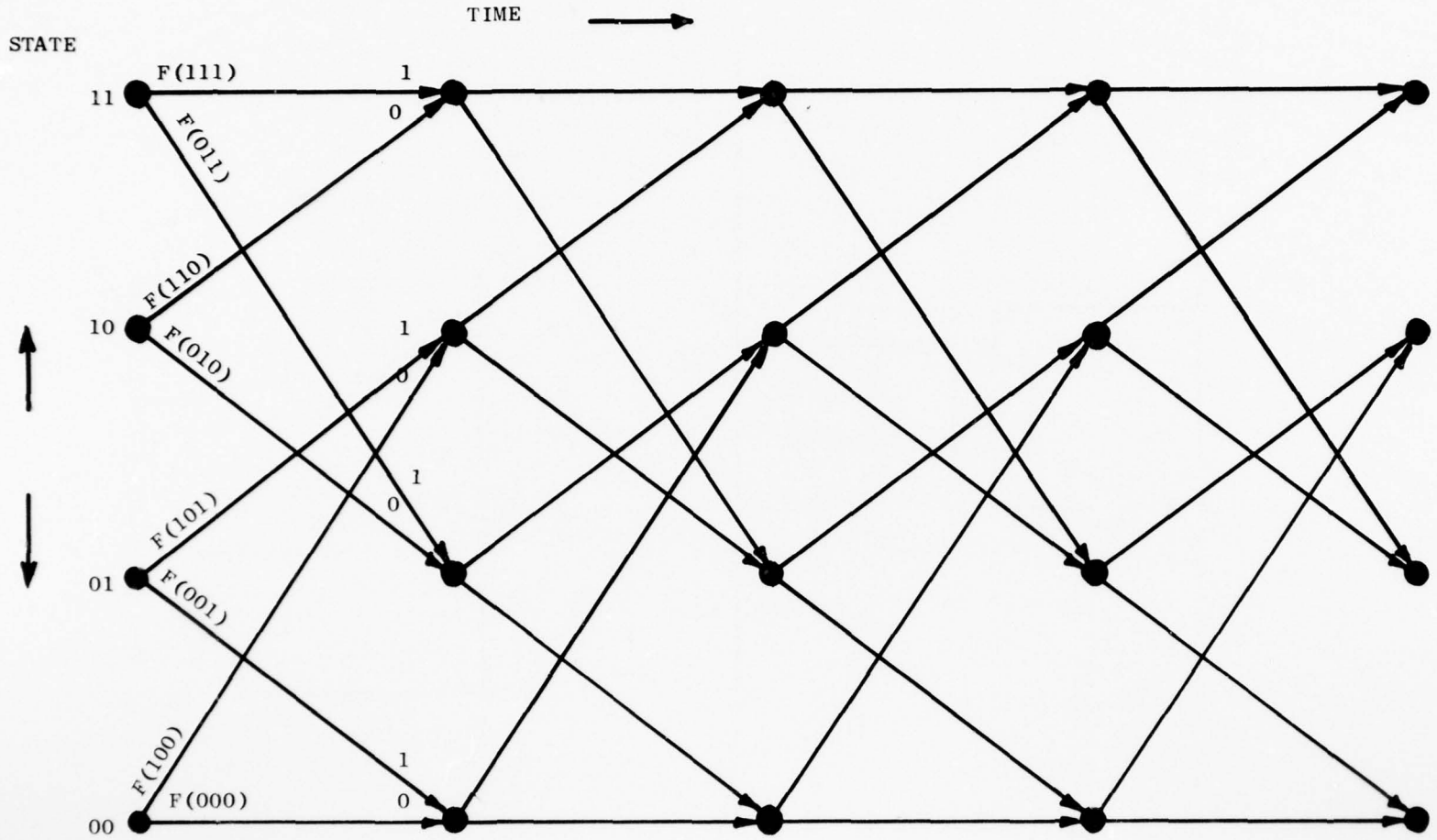


Fig. 3 A TRELLIS MATCHED TO A LENGTH 3 DECODER

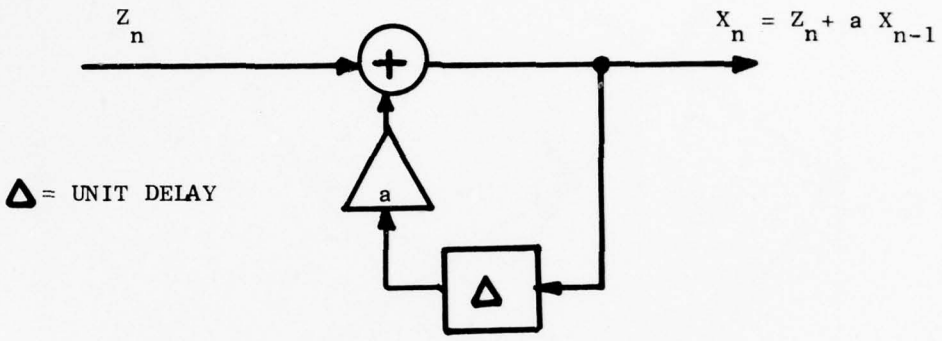


Fig. 4 AUTOREGRESSIVE FILTER

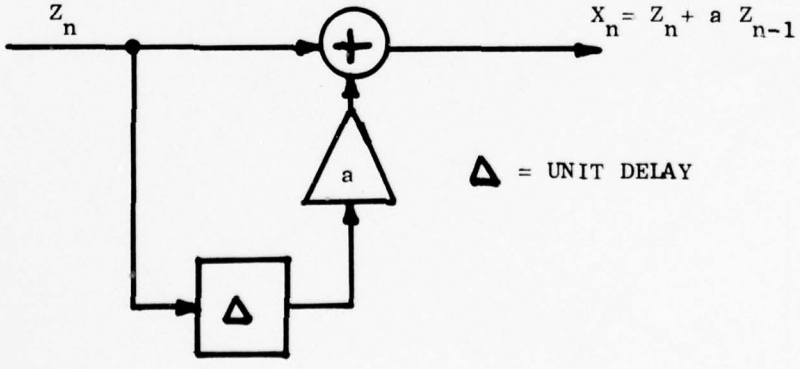
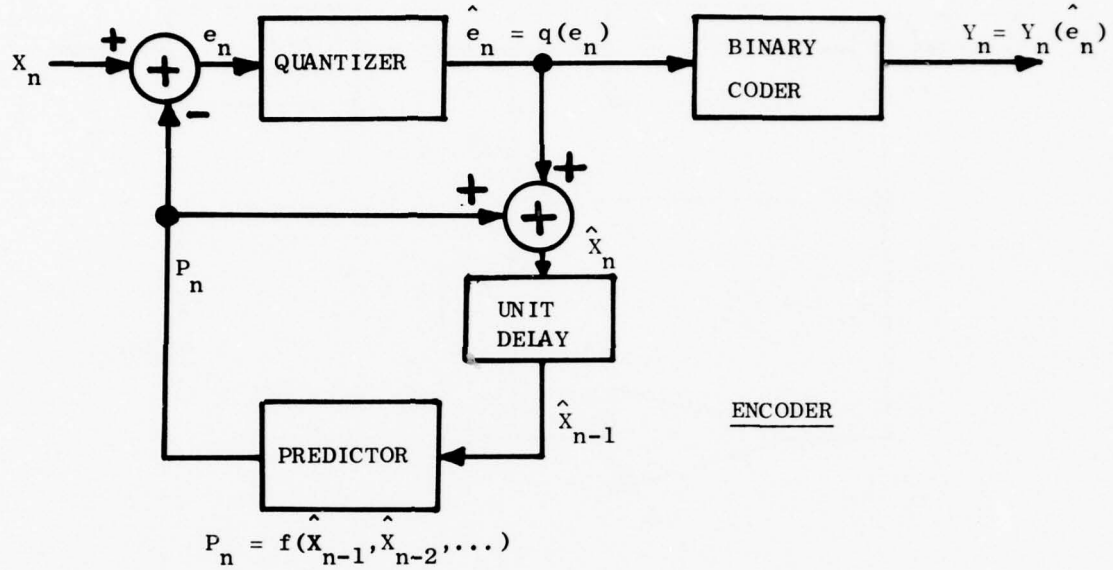


Fig. 5 MOVING AVERAGE FILTER



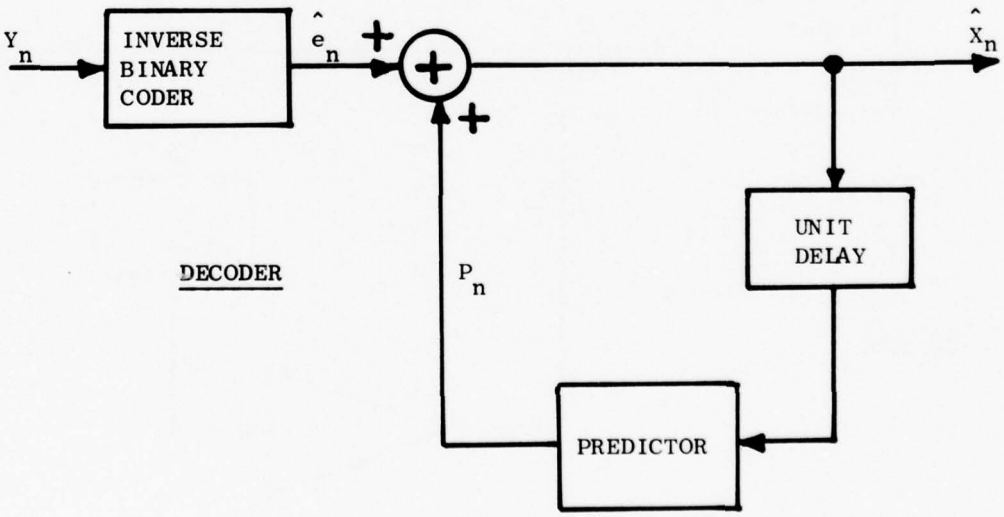


Fig. 6 PREDICTIVE QUANTIZATION SYSTEM

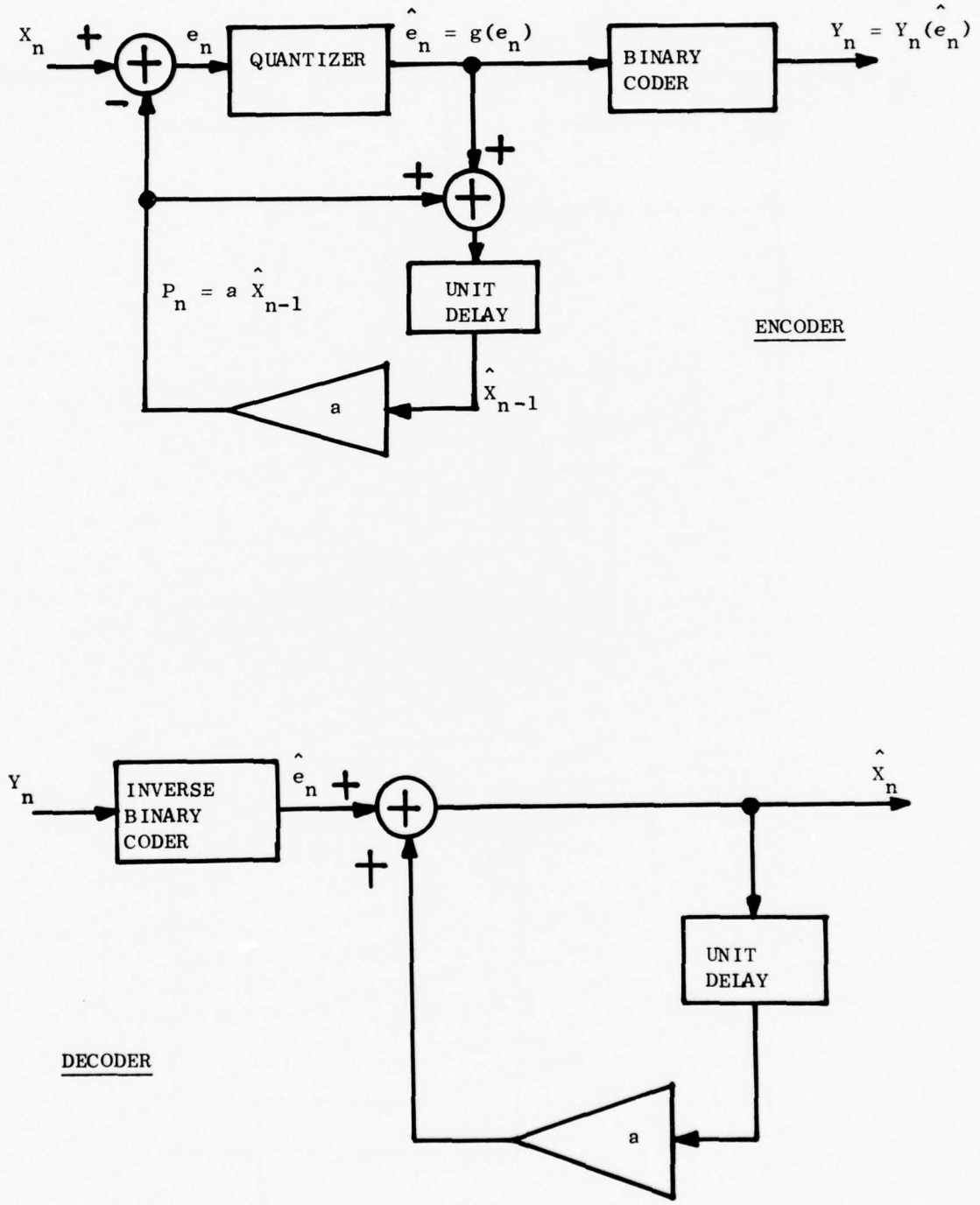
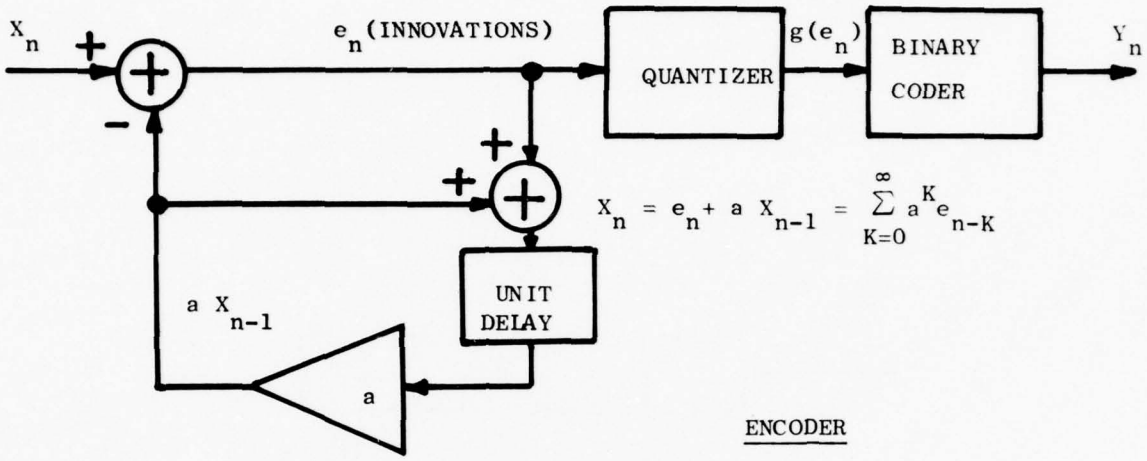


Fig. 7 LINEAR PREDICTIVE QUANTIZER



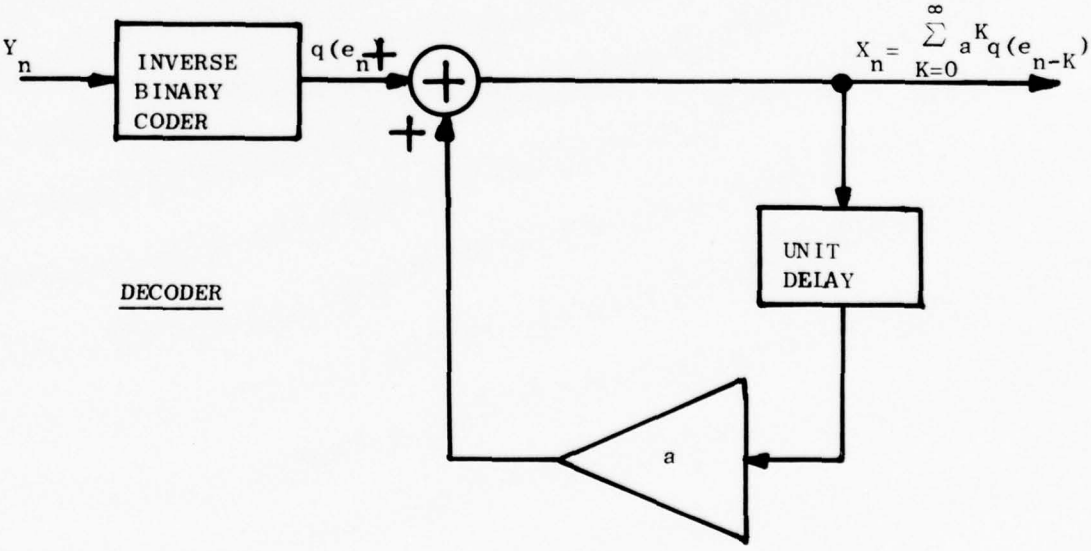


Fig. 8 INNOVATION QUANTIZER

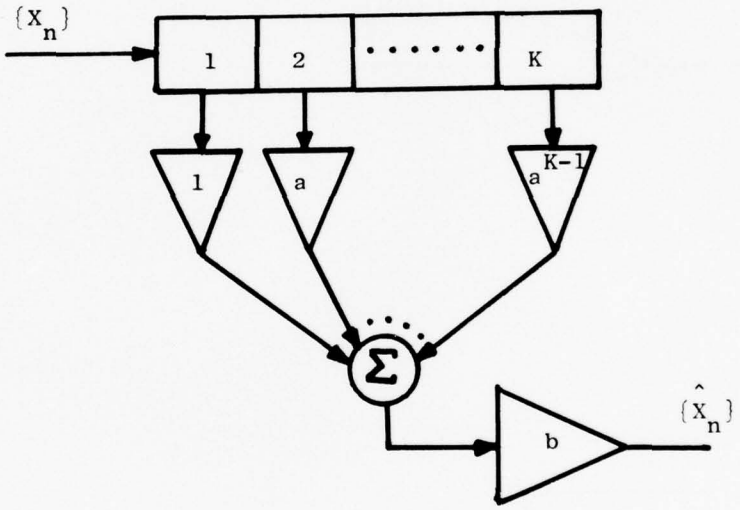


Fig. 9 THE TRUNCATED PREDICTIVE QUANTIZER DECODER

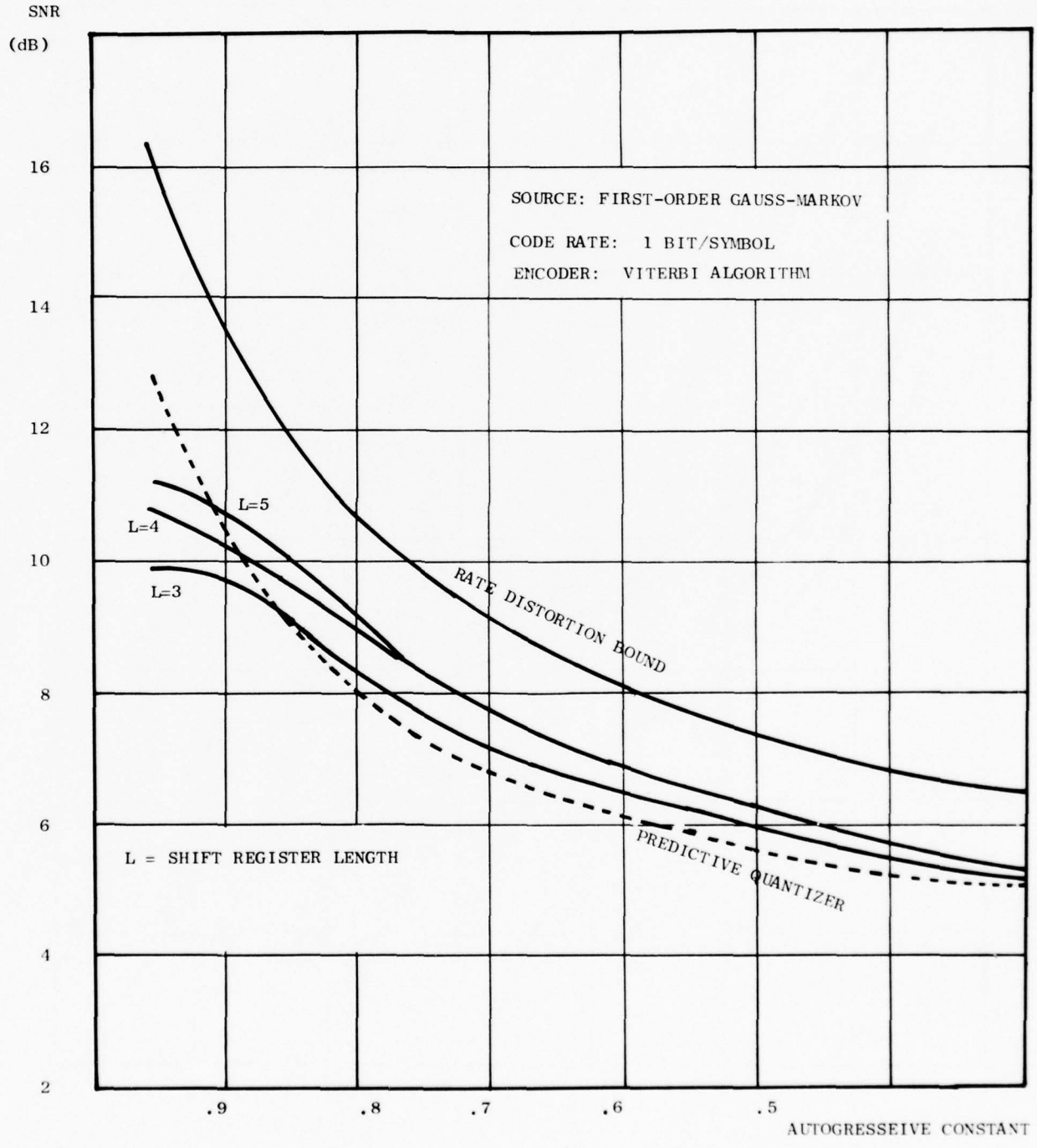


Fig. 10 PERFORMANCE OF TRUNCATED PREDICTIVE QUANTIZER DECODER

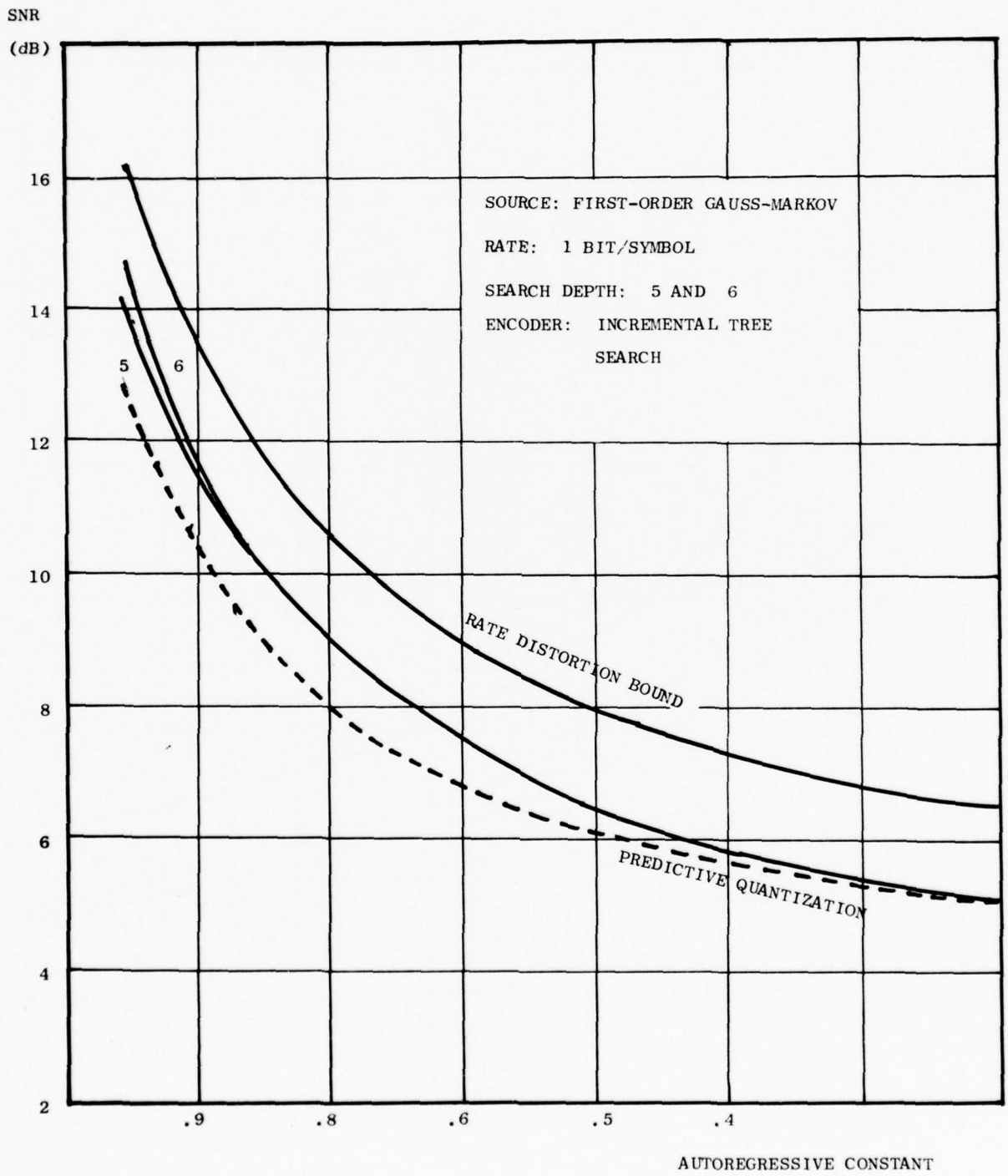


Fig. 11 LOOK AHEAD PREDICTIVE QUANTIZATION

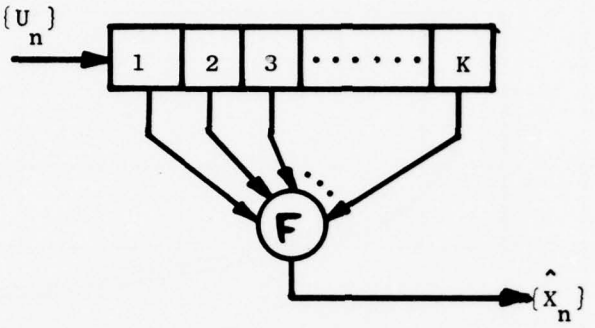


Fig. 12 THE FAKE PROCESS

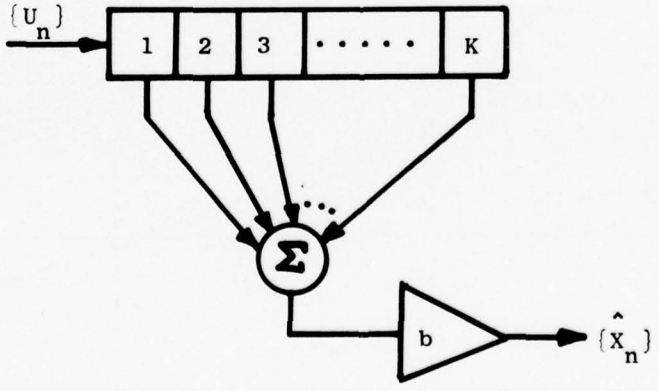


Fig. 13 THE C.L.T. DECODER

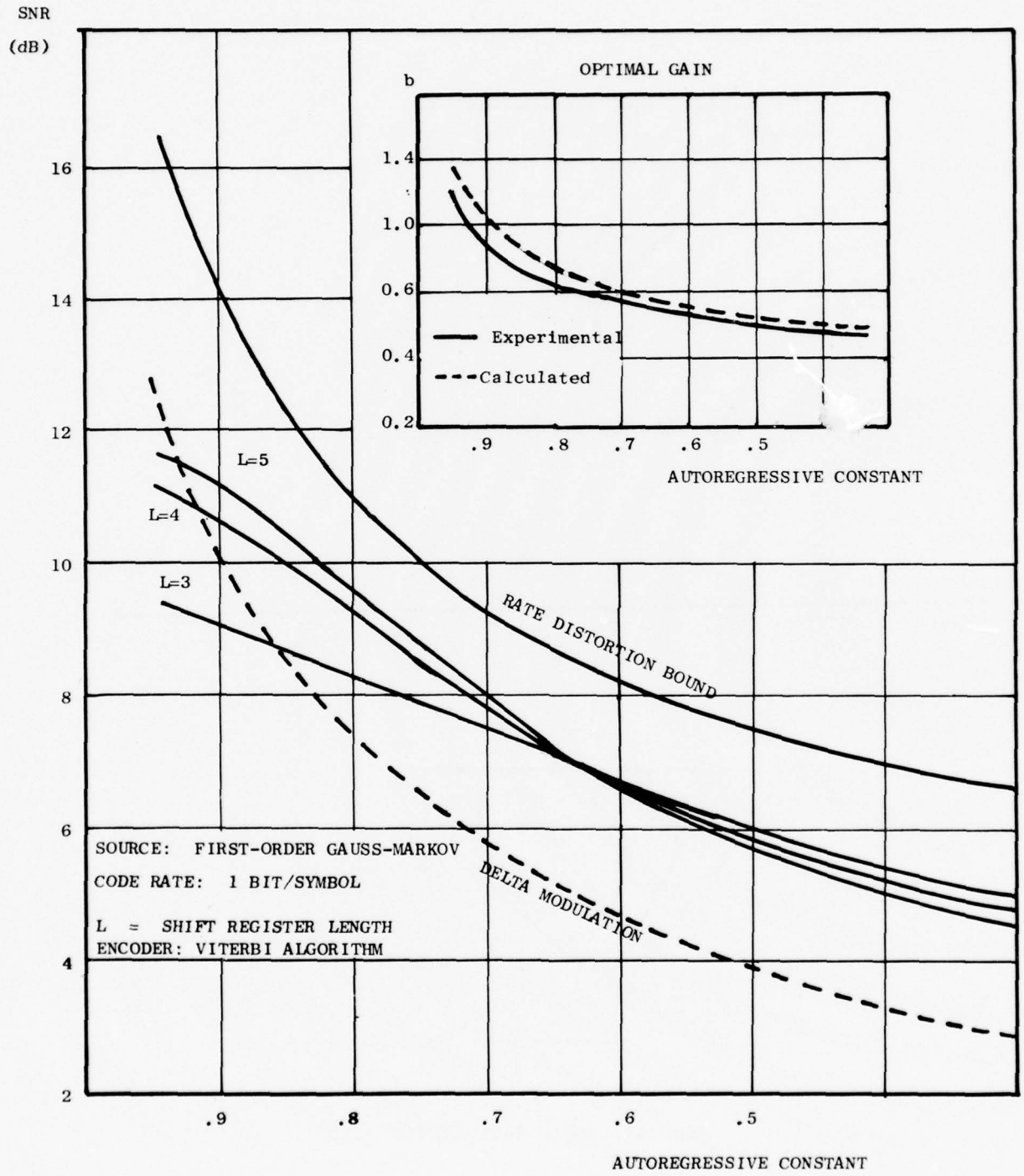


Fig. 14 PERFORMANCE OF THE C.L.T. DECODER

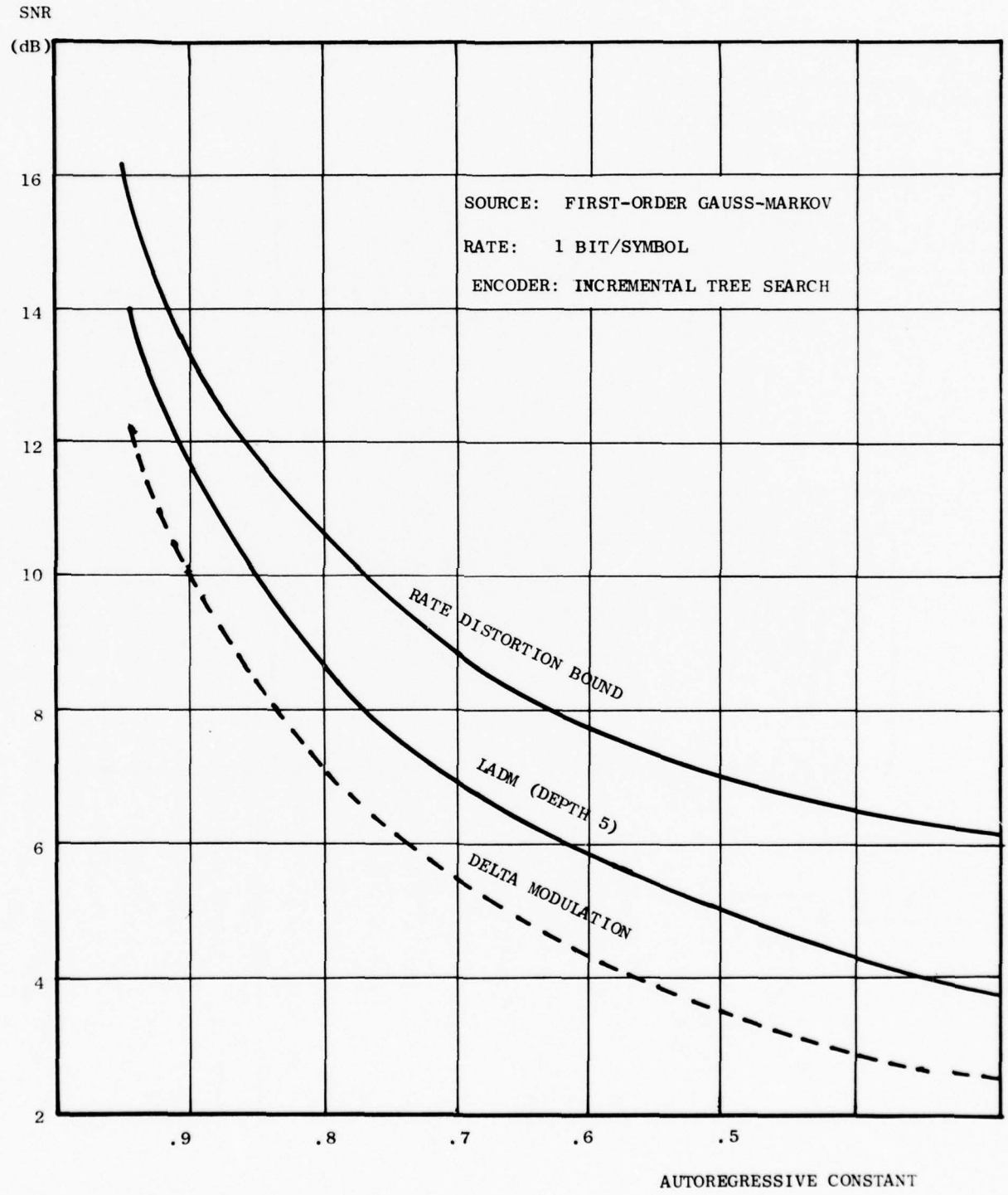
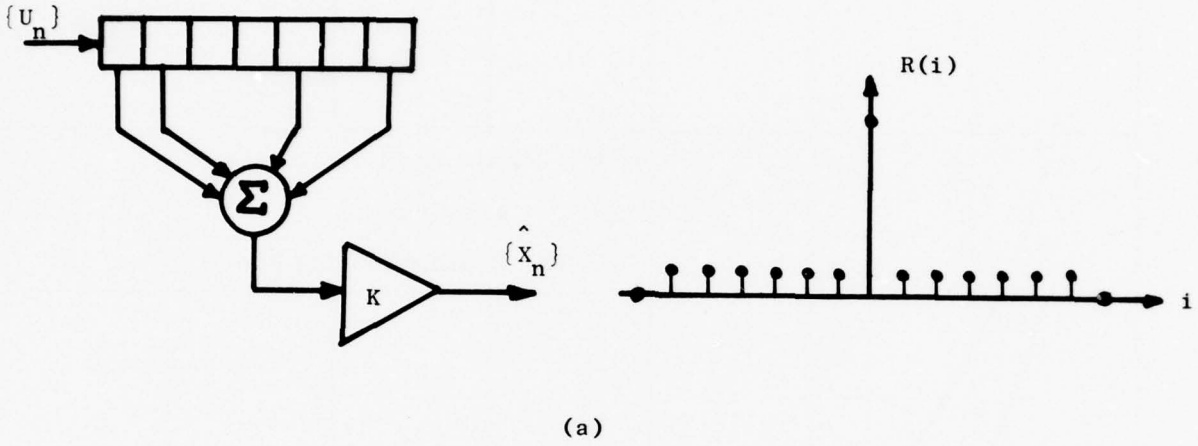
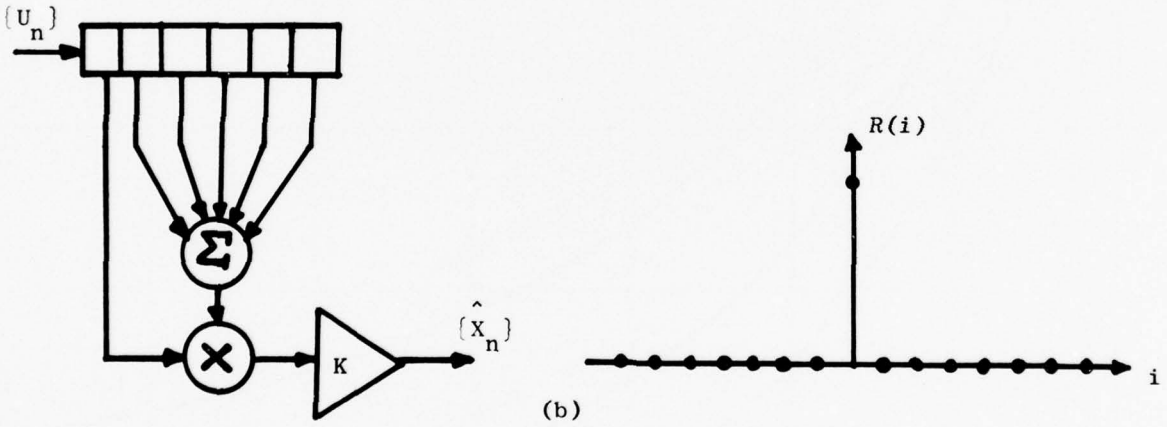


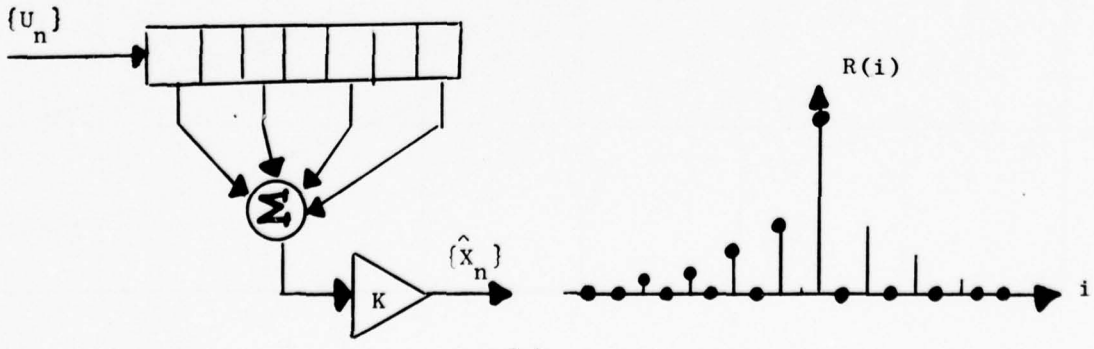
Fig. 15 LOOK AHEAD DELTA MODULATION



(a)



(b)



(c)

Fig. 16 Modified C.L.T. Decoders

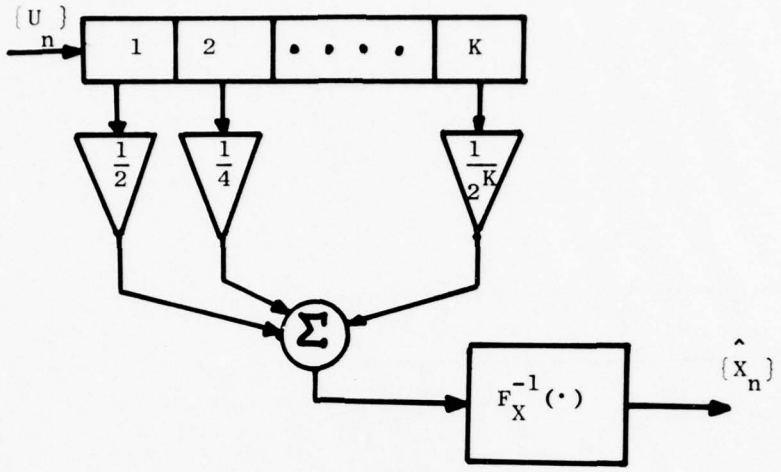


Fig. 17 INVERSE DISTRIBUTION DECODER

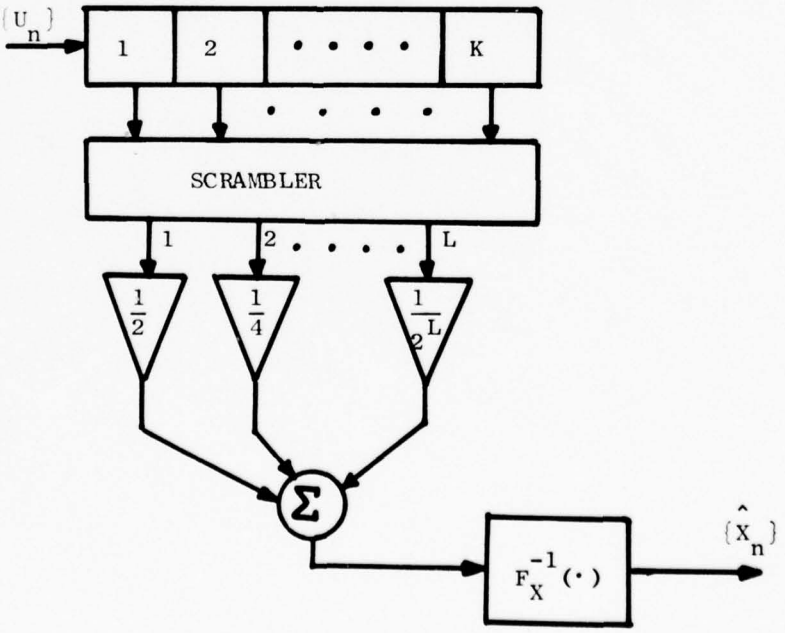


Fig. 18 BLOCK SCRAMBLER

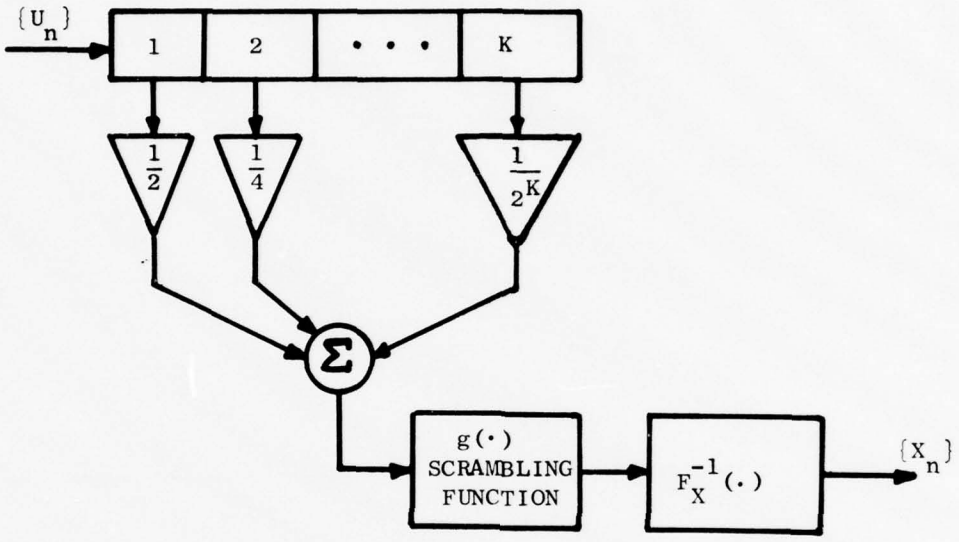


Fig. 19 SCRAMBLING FUNCTION SCRAMBLER

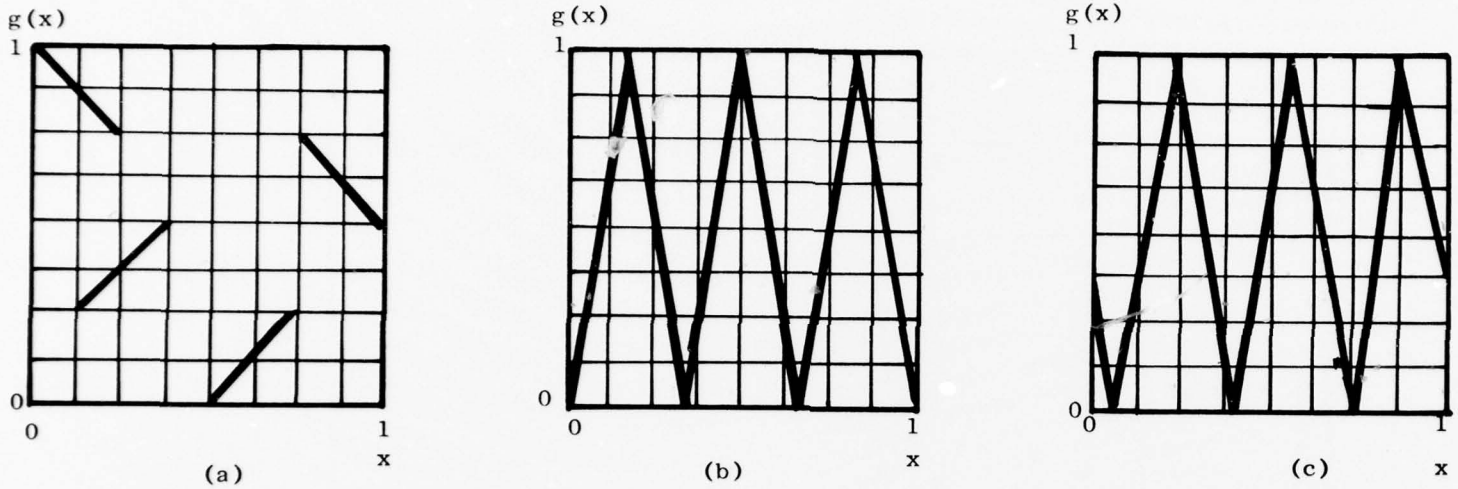


Fig. 20 EXAMPLES OF SCRAMBLING FUNCTIONS

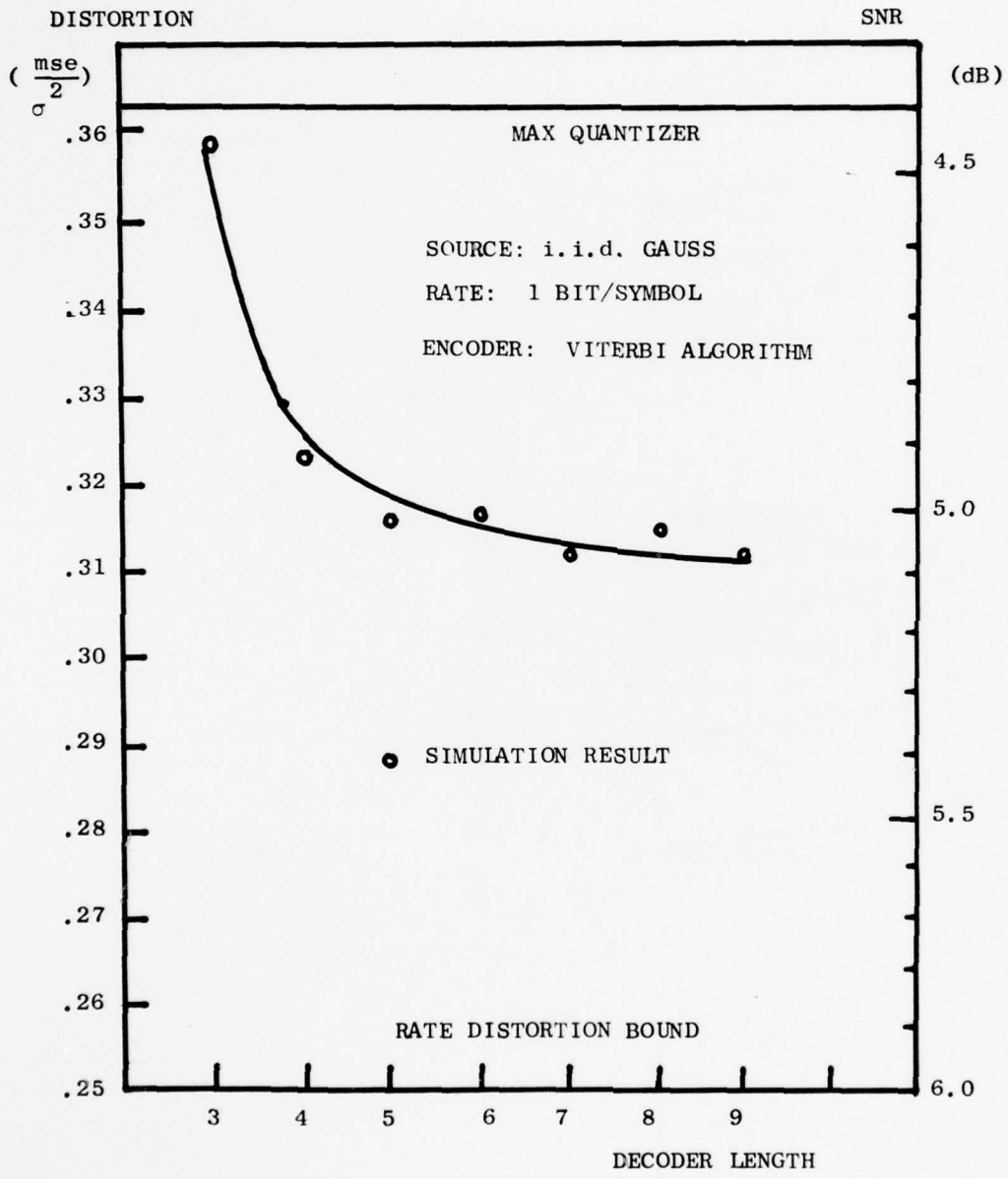
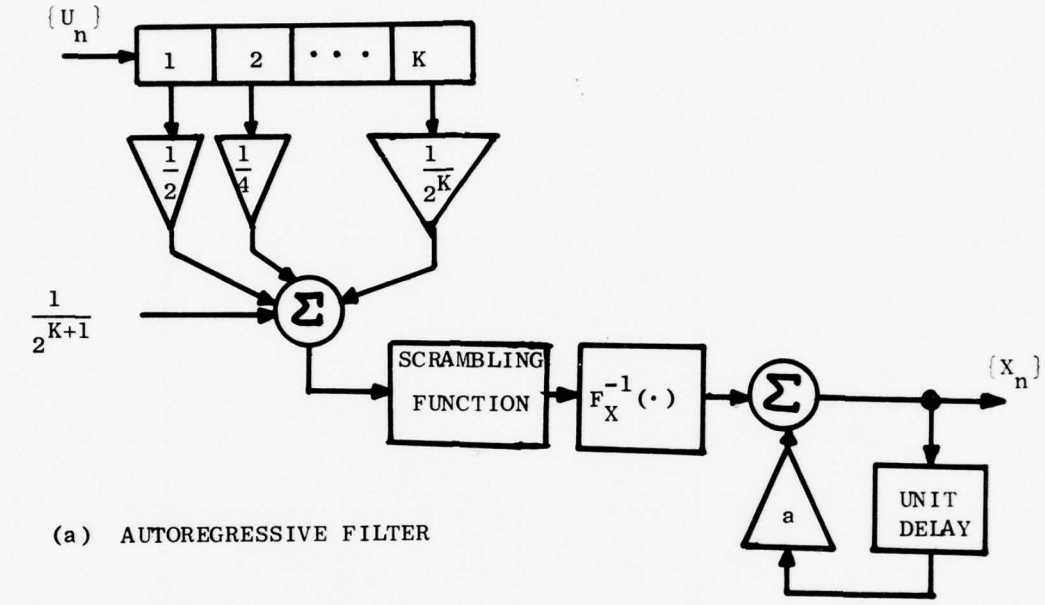
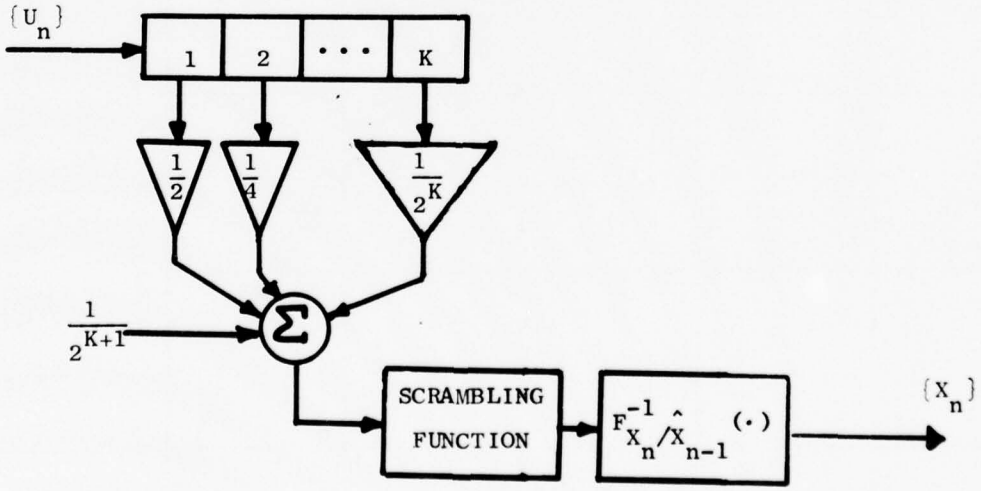


Fig. 21 PERFORMANCE: INVERSE DISTRIBUTION/SCRAMBLING FUNCTION DECODER



(a) AUTOREGRESSIVE FILTER



(b) CONDITIONAL INVERSE DISTRIBUTION

Fig. 22 FAKE AUTOREGRESSIVE PROCESSES

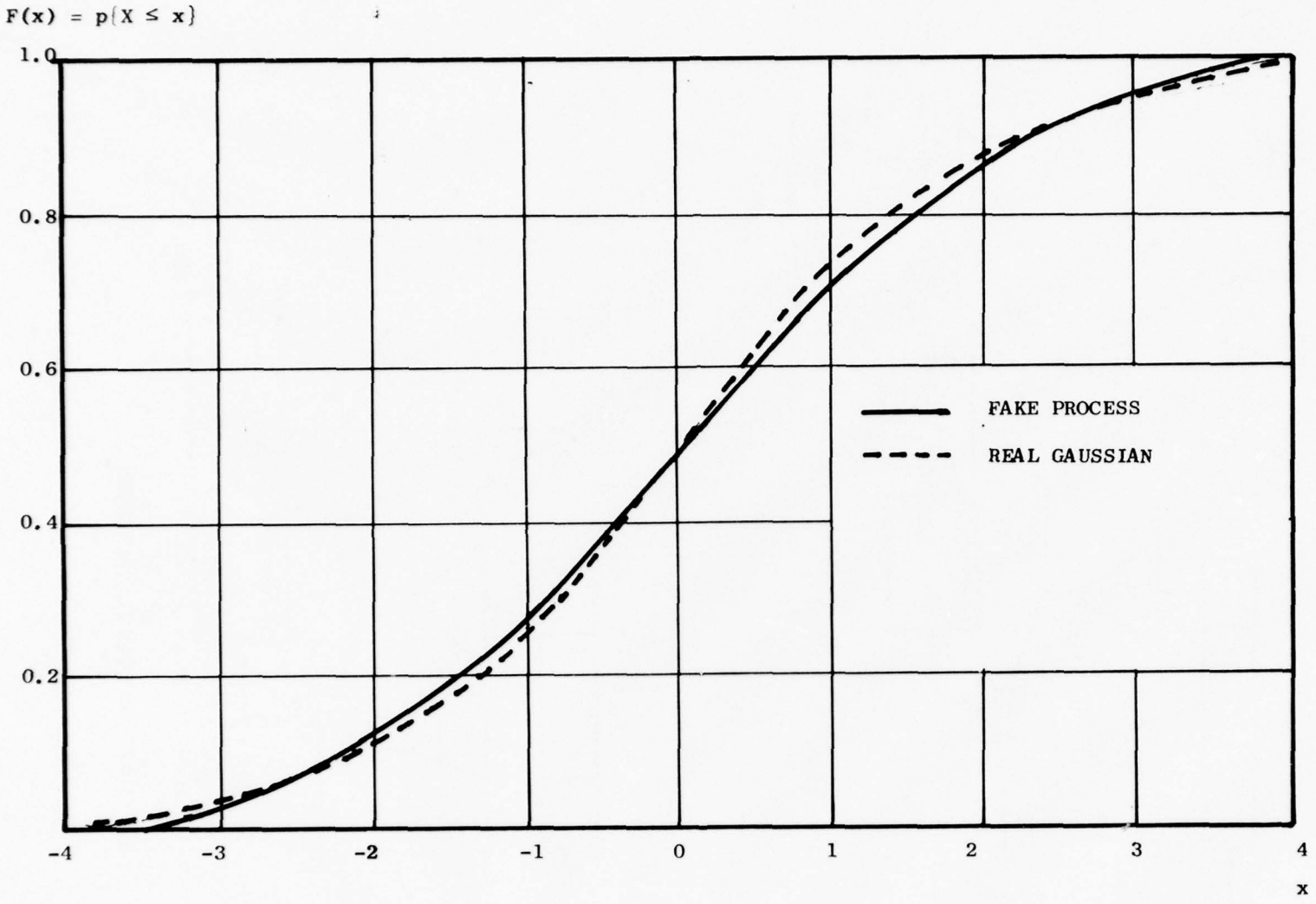


Fig. 23 DISTRIBUTION FUNCTIONS OF REAL GAUSSIAN AND FAKE PROCESSES

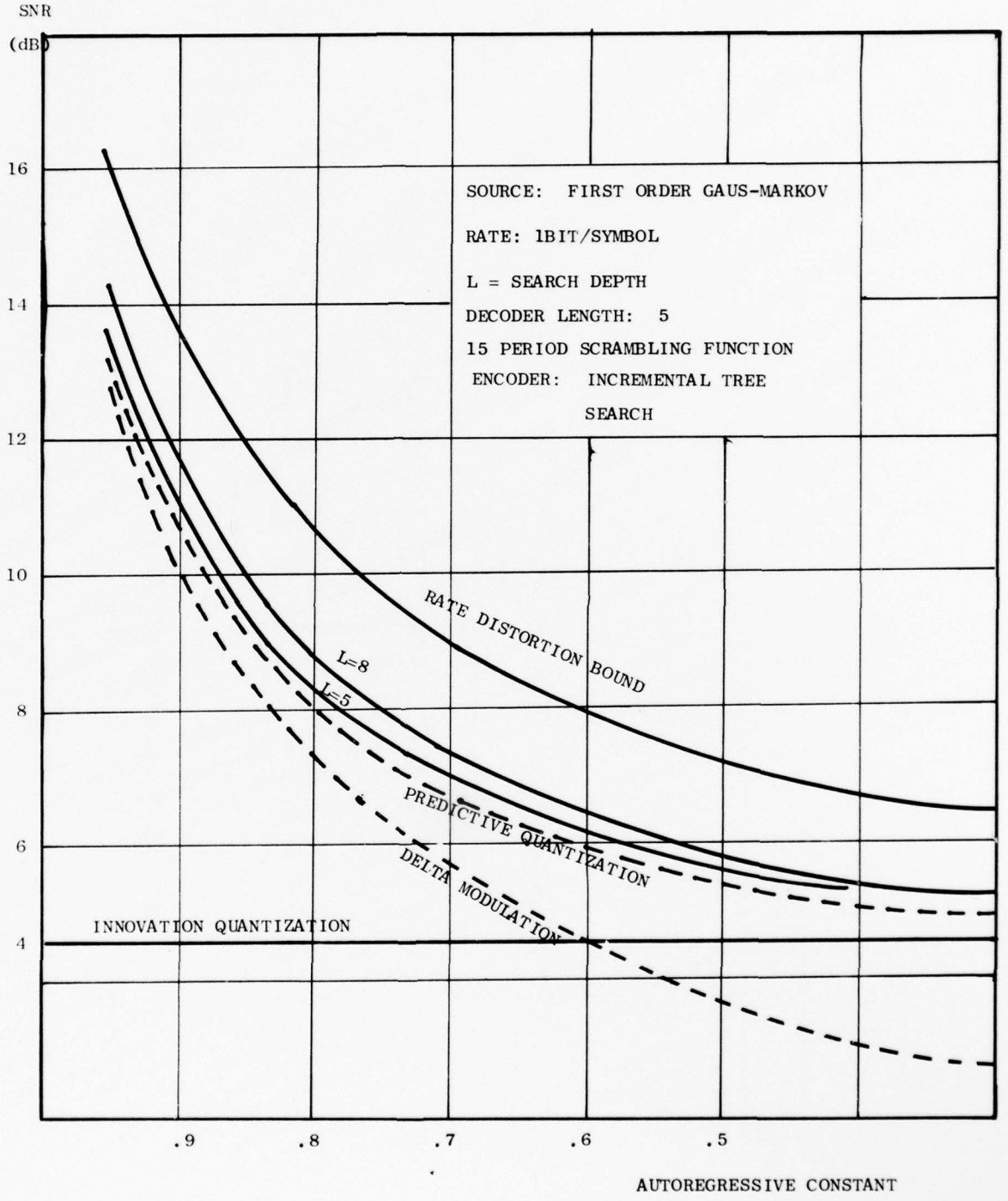


Fig. 24 PERFORMANCE: FAKE AUTOREGRESSIVE DECODER

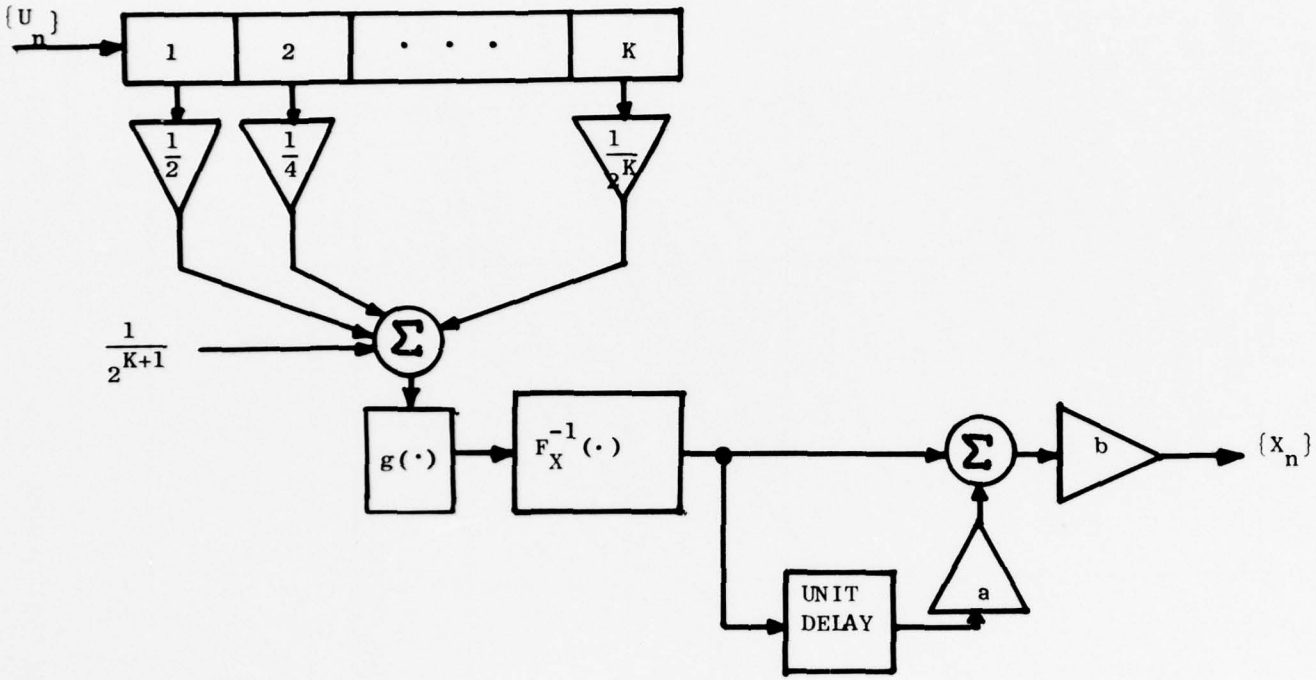


Fig. 25 FAKE MOVING AVERAGE PROCESS

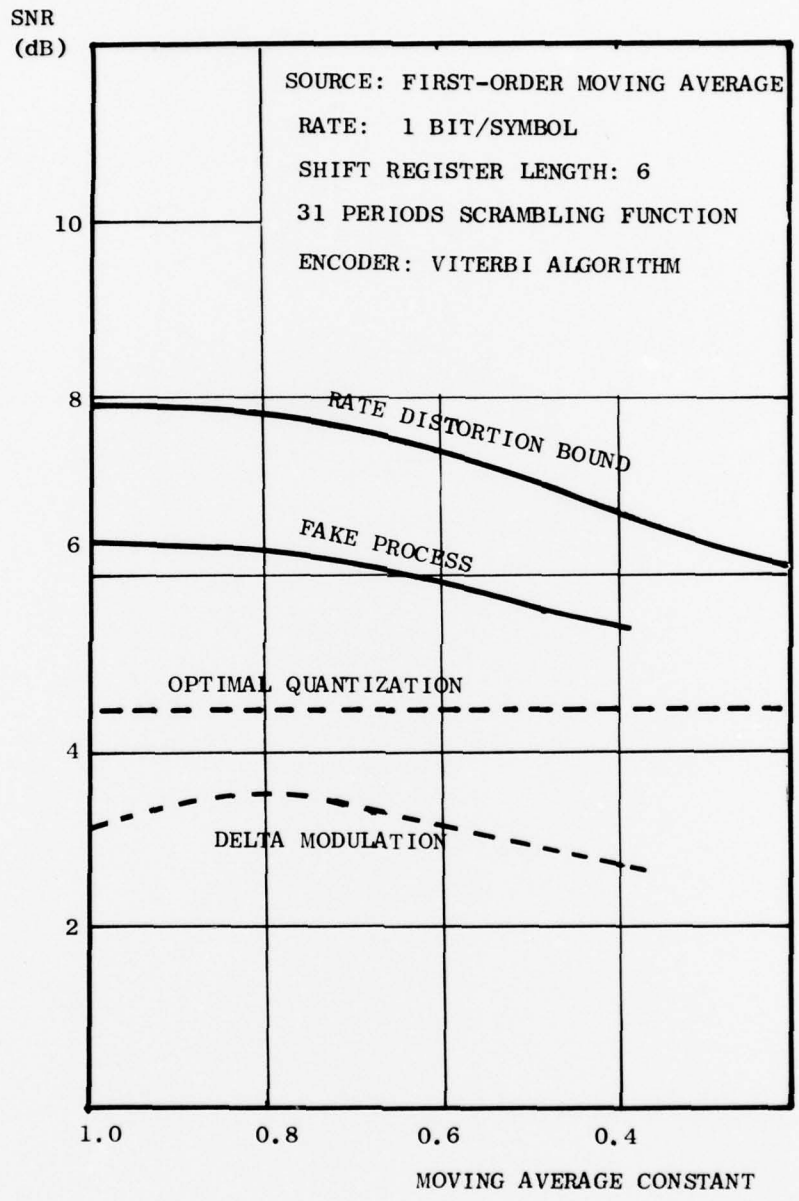


Fig. 26 PERFORMANCE: FAKE MOVING AVERAGE PROCESS

REFERENCES

[1] Toby Berger, Rate Distortion Theory - A Mathematical Basis for Data Compression, Prentice Hall Inc. New Jersey, 1971.

[2] Robert G. Gallager, Information Theory and Reliable Communication, John Wiley & Sons, Inc., New York, 1968 (Chapter 9).

[3] Lee D. Davisson and Robert M. Gray, editors, Data Compression, (Benchmark Papers in Electrical Engineering and Computer Science, Vol. 14), Dowden, Hutchinson & Ross, Inc., Stroudsbourg, Penn., 1976.

[4] Herbert Gish and John H. Pierce, "Asymptotically Efficient Quantizing," IEEE Trans. Info. They., IT-15 (2), pp. 248-252, 1969.

[5] T.J. Goblick, Jr. and J.L. Holsinger, "Analog Source Digitization: A Comparison of Theory and Practice," IEEE Trans. Info. They., (Correspondence), Vol. IT-13, pp. 323-326, April 1967.

[6] F. Jelinek, "Tree Encoding of Memoryless Time-Discrete Sources with a Fidelity Criterion," IEEE Trans. Info. They., Vol. IT-15, pp. 584-590, Sept. 1969.

[7] C.R. Davis and M.E. Hellman, "On Tree Coding with a Fidelity Criteria," IEEE Trans. Info. They., IT-21, pp. 373-378, July 1975.

[8] R.G. Gallager, "Tree Encoding for Sources with a Distortion Measure," IEEE Trans. Info. They., Vol. IT-20, pp. 65-76, Jan. 1974.

[9] A.J. Viterbi and J.K. Omura, "Trellis Encoding of Memoryless-Discrete-Time Sources with a Fidelity Criteria," IEEE Trans. Info. They., IT-20 (3), pp. 325-332, 1974.

[10] R.M. Gray, "Time-Invariant Trellis Encoding of Ergodic Discrete-Time Sources with a Fidelity Criterion," IEEE Trans. Info. They., Vol. IT-23, pp. 71-83, Jan. 1977.

[11] R.M. Gray, D.L. Neuhoff and D.S. Ornstein, "Nonblock Source Coding with a Fidelity Criterion," Annals of Prob. Vol. 3, pp. 478-491, April 1975.

[12] R.M. Gray, "Sliding-Block Source Coding," IEEE Trans. Info. They., Vol. IT-21, pp. 357-367, July 1975.

[13] A.J. Viterbi, "Error Bounds for Convolutional Codes and An Asymptotically Optimum Decoding Algorithm," IEEE Trans. Info. They., Vol. IT-13, pp. 260-269, April 1967.

[14] G.D. Forney, Jr., "The Viterbi Algorithm," Proc. IEEE, Vol. 61, pp. 268-278, March 1973.

[15] F. Jelinek and J.B. Anderson, "Instrumentable Tree Encoding of Information Sources," IEEE Trans. Info. They., Vol. IT-17, pp. 118-119, Jan. 1971.

[16] S.G. Wilson and D.W. Lytle, "Trellis Encoding of Continuous-Amplitude Memoryless Sources," IEEE Trans. Info. They., Vol. IT-23, pp. 404-409, May 1977.

[17] J.W. Mark, "Adaptive Trellis Encoding of Discrete-Time Sources with a Distortion Measure," IEEE Trans. Commun., Vol. COM-25, pp. 408-417, April 1977.

[18] J.B. Anderson and J.B. Bodie, "Tree Encoding of Speech," IEEE Trans. Info. They., IT-21, pp. 379-381, July 1975.

[19] J. Uddenfeld and L.H. Zettenberg, "Algorithms for Delayed Encoding in Delta Modulation with Speech-Like Signals," IEEE Trans. Commun., Vol. COM-24, pp. 652-658, June 1976.

[20] L.H. Zettenberg and J. Uddenfeld, "Adaptive Delta Modulation with Delayed Decision," IEEE Trans. Commun., Vol. COM-22, pp. 1195-1198, Sept. 1974.

[21] R.M. Gray, "Information Rates of Autoregressive Sources," IEEE Trans. Info. They., IT-18, pp. 412-421, July 1970.

[22] R.M. Gray and D.S. Ornstein, "Sliding-Block Joint Source/Noisy Channel Coding Theorems," IEEE Trans. Info. They., Vol. IT-22, pp. 682-690, Nov. 1976.

[23] D.S. Arnstein, "Quantization Error in Predictive Coders," IEEE Trans. Commun., Vol. COM-23, pp. 423-429, April 1975.

[24] R.M. Gray, D.L. Neuhoff and P.C. Shields, "A Generalization of Ornstein's $\bar{d}$-distance with Application to Information Theory," Ann. Prob., Vol. 3, pp. 315-328, 1975.

[25] R.M. Gray, D.L. Neuhoff and J.K. Omura, "Process Definitions of Distortion-Rate Functions and Source Coding Theorems," IEEE Trans. Info. They., Vol. IT-21, No. 5, pp. 524-532, 1975.

[26] J. Linde, Ph.D. Dissertation, Stanford University, Stanford, California, 1977.

[27] J. Max, "Quantizing for Minimum Distortion," IRE Trans. Info. They., 6, pp. 7-12, 1960.

[28] A. Papoulis, Probability, Random Variables and Stochastic Processes, McGraw-Hill, New York, 1965 (Section 11-6).

[29] R.M. Fano, "A Heuristic Discussion of Probabilistic Decoding," IEEE Trans. on Info. They., Vol. IT-19, pp. 64-73, 1973.

[30] F. Jelinek, "A Fast Sequential Decoding Algorithm Using a Stack," IBM J. of Res. and Dev., Vol. 13, pp. 675-685, 1969.

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 6504-2	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* The Design of Tree and Trellis Data Compression Systems		5. TYPE OF REPORT & PERIOD COVERED ISL Technical Report 2/77 - 2/78
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR*(s)* Joseph Linde and Robert M. Gray		8. CONTRACT OR GRANT NUMBER*(s)* AFOSR F44620-73-C-0065
9. PERFORMING ORGANIZATION NAME AND ADDRESS Stanford University Electrical Engineering Dept. Stanford, CA 94305		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS AFOSR/NM Bolling AFB, Bldg. 410 Washington, D.C. 20332		12. REPORT DATE February 1978 13. NUMBER OF PAGES 75
14. MONITORING AGENCY NAME & ADDRESS*(if different from Controlling Office)*		15. SECURITY CLASS. *(of this report)* Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*
This document has been approved for public release and sale; its distribution is unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*
Tree codes, trellis codes, data compression, fake process

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*
Recent results in information theory promise the existence of tree and trellis data compression systems operating near the Shannon theoretical bound, but provide little indication of how actually to design such systems. We here present several intuitive design approaches and also a general design philosophy based upon the generation of "fake processes" i.e., finite entropy processes which are close (in the generalized Ornstein distance) to the process one wishes to compress. Most of the design procedures can be used for a wide class of sources. Performance is evaluated, via simulations, for memoryless, autoregressive and

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

cont. next page

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

20 cont.

moving average Gaussian sources and comparedto traditional Data Compression systems. The new schemes typically provide 1-2 dB improvement in performance over the traditional schemes at a rate of 1 bit/symbol. The inevitable increase in complexity is moderate in most cases.

JSEP REPORTS DISTRIBUTION LIST

Department of Defense

Director
National Security Agency
Attn: Dr. T. J. Beahn
Fort George G. Meade
Maryland 20755

Defense Documentation Center (12)
Attn: DDC-TCA (Mrs. V. Caponio)
Cameron Station
Alexandria, Virginia 22314

Assistant Director
Electronics and Computer Sciences
Office of Director of Defense
Research and Engineering
The Pentagon
Washington, D.C. 20315

Defense Advanced Research
Projects Agency
Attn: Dr. R. Reynolds
1400 Wilson Boulevard
Arlington, Virginia 22209

Department of the Army

Commandant
US Army Air Defense School
Attn: ATSAD-T-CSM
Fort Bliss, Texas 79916

Commander
US Army Armament R&D Command
Attn: DRDAR-TSS
Dover, New Jersey 07801

Commander
US Army Armament R&D Command (BRL)
Attn: DRDAR-TSB-S
Aberdeen Proving Ground
Aberdeen, Maryland 21005

Commandant
US Army Command and General Staff
College
Attn: Acquisitions, Library Division
Fort Leavenworth, Kansas 66027

Commander
US Army Communication Command
Attn: CC-OPS-PD
Fort Huachuca, Arizona 85613

Commander
US Army Materials and Mechanics
Research Center
Attn: Chief, Materials Science Div.
Watertown, Massachusetts 02172

Commander
US Army Materiel Development and
Readiness Command
Attn: Technical Library, Rm. 7S 35
5001 Eisenhower Avenue
Alexandria, Virginia 22333

Commander
US Army Missile R&D Command
Attn: Chief, Document Section
Redstone Arsenal, Alabama 35809

Commander
US Army Satellite Communications Agency
Fort Monmouth, New Jersey 07703

Director
US Army Signals Warfare Laboratory
Attn: DELSW-OS
Arlington Hall Station
Arlington, Virginia 22212

Project Manager
ARTADS
EAI Building
West Long Branch, New Jersey 07764

NOTE: One (1) copy to each addressee unless otherwise indicated.

Commander/Director
Atmospheric Sciences Lab. (ECOM)
Attn: DRSEL-BL-DD
White Sands Missile Range
New Mexico 88002

Commander
US Army Electronics Command
Attn: DRSEL-NL-O (Dr. H. S. Bennett)
Fort Monmouth, New Jersey 07703

Director
TRI-TAC
Attn: TT-AD (Mrs. Briller)
Fort Monmouth, New Jersey 07703

Commander
US Army Electronics Command
Attn: DRSEL-CT-L (Dr. R. Buser)
Fort Monmouth, New Jersey 07703

Director
Electronic Warfare Lab. (ECOM)
Attn: DRSEL-WL-MY
White Sands Missile Range
New Mexico 88002

Executive Secretary, TAC/JSEP
US Army Research Office
P. O. Box 12211
Research Triangle Park
North Carolina 27709

Commander
Harry Diamond Laboratories
Attn: Mr. John E. Rosenberg
2800 Powder Mill Road
Adelphi, Maryland 20783

HQDA (DAMA-ARZ-A)
Washington, D.C. 20310

Commander
US Army Electronics Command
Attn: DRSEL-TL-E (Dr. J. A. Kohn)
Fort Monmouth, New Jersey 07703

Commander
US Army Electronics Command
Attn: DRSEL-TL-EN
(Dr. S. Kroenenberg)
Fort Monmouth, New Jersey 07703

Commander
US Army Electronics Command
Attn: DRSEL-NL-T (Mr. R. Kulinyi)
Fort Monmouth, New Jersey 07703

Commander
US Army Electronics Command
Attn: DRSEL-NL-B (Dr. E. Lieblein)
Fort Monmouth, New Jersey 07703

Commander
US Army Electronics Command
Attn: DRSEL-TL-MM (Mr. N. Lipetz)
Fort Monmouth, New Jersey 07703

Commander
US Army Electronics Command
Attn: DRSEL-RD-O (Dr. W. S. McAfee)
Fort Monmouth, New Jersey 07703

Director
Night Vision Laboratory
Attn: DRSEL-NV-D
Fort Belvoir, Virginia 22060

Col. Robert Noce
Senior Standardization Representative
US Army Standardization Group, Canada
Canadian Force Headquarters
Ottawa, Ontario, CANADA KIA)K2

Commander
US Army Electronics Command
Attn: DRSEL-NL-B (Dr. D. C. Pearce)
Fort Monmouth, New Jersey 07703

Commander
US Army Electronics Command
Attn: DRSEL-NL-RH-1
(Dr. F. Schwering)
Fort Monmouth, New Jersey 07703

Commander
US Army Electronics Command
Attn: DRSEL-TL-I
(Dr. C. G. Thornton)
Fort Monmouth, New Jersey 07703

US Army Research Office (3)
Attn: Library
P. O. Box 12211
Research Triangle Park
North Carolina 27709

Director
Division of Neuropsychiatry
Walter Reed Army Institute
of Research
Washington, D.C. 20012

Commander
White Sands Missile Range
Attn: STEWS-ID-R
White Sands Missle Range
New Mexico 88002

Department of the Air Force

Mr. Robert Barrett
RADC/ETS
Hanscom AFB, Massachusetts 01731

Dr. Carl E. Baum
AFWL (ES)
Kirtland AFB, New Mexico 87117

Dr. E. Champagne
AFAL/DH
Wright-Patterson AFB, Ohio 45433

Dr. R. P. Dolan
RADC/ETSD
Hanscom AFB, Massachusetts 01731

Mr. W. Edwards
AFAL/TE
Wright-Patterson AFB, Ohio 45433

Professor R. E. Fontana
Head, Department of Electrical
Engineering
AFIT/ENE
Wright-Patterson AFB, Ohio 45433

Dr. Alan Garscadden
AFAPL/POD
Wright-Patterson AFB, Ohio 45433

USAF European Office of Aerospace
Research
Attn: Major J. Gorrell
Box 14, FPO, New York 09510

LTC Richard J. Gowen
Department of Electrical Engineering
USAF Academy, Colorado 80840

Mr. Murray Kesselman (ISCA)
Rome Air Development Center
Griffiss AFB, New York 13441

Dr. G. Knausenberger
Air Force Member, TAC
Air Force Office of Scientific
Research, (AFSC) AFSOR/NE
Bolling Air Force Base, DC 20332

Dr. L. Kravitz
Air Force Member, TAC
Air Force Office of Scientific
Research, (AFSC) AFSOR/NE
Bolling Air Force Base, DC 20332

Mr. R. D. Larson
AFAL/DHR
Wright-Patterson AFB, Ohio 45433

Dr. Richard B. Mack
RADC/ETER
Hanscom AFB, Massachusetts 01731

Mr. John Mottsmith (MCIT)
HQ ESD (AFSC)
Hanscom AFB, Massachusetts 01731

Dr. Richard Picard
RADC/ETSL
Hanscom AFB, Massachusetts 01731

Dr. J. Ryles
Chief Scientist
AFAL/CA
Wright-Patterson AFB, Ohio 45433

Dr. Allan Schell
RADC/ETE
Hanscom AFB, Massachusetts 01731

Mr. H. E. Webb, Jr. (ISCP)
Rome Air Development Center
Griffisss AFB, New York 13441

LTC G. Wepfer
Air Force Office of Scientific
Research, (AFSC) AFOSR/NP
Bolling Air Force Base, DC 20332

LTC G. McKemie
Air Force Office of Scientific
Research, (AFSC) AFOSR/NM
Bolling Air Force Base, DC 20332

Department of the Navy

Dr. R. S. Allgaier
Naval Surface Weapons Center
Code WR-303
White Oak
Silver Spring, Maryland 20910

Naval Weapons Center
Attn: Code 5515, H. F. Blazek
China Lake, California 93555

Dr. H. L. Blood
Technical Director
Naval Undersea Center
San Diego, California 95152

Naval Research Laboratory
Attn: Code 5200, A. Brodzinsky
4555 Overlook Avenue, SW
Washington, D.C. 20375

Naval Research Laboratory
Attn: Code 7701, J. D. Brown
4555 Overlook Avenue, SW
Washington, D.C. 20375

Naval Research Laboratory
Attn: Code 5210, J. E. Davey
4555 Overlook Avenue, SW
Washington, D.C. 20375

Naval Research Laboratory
Attn: Code 5460/5410, J. R. Davis
4555 Overlook Avenue, SW
Washington, D.C. 20375

Naval Ocean Systems Center
Attn: Code 75, W. J. Dejka
271 Catalina Boulevard
San Diego, California 92152

Naval Weapons Center
Attn: Code 601, F. C. Essig
China Lake, California 93555

Naval Research Laboratory
Attn: Code 5510, W. L. Faust
4555 Overlook Avenue, SW
Washington, D.C. 20375

Naval Research Laboratory
Attn: Code 2626, Mrs. D. Folen
4555 Overlook Avenue, SW
Washington, D.C. 20375

Dr. Robert R. Fossum
Dean of Research
Naval Postgraduate School
Monterey, California 93940

Dr. G. G. Gould
Technical Director
Naval Coastal System Laboratory
Panama City, Florida 32401

Naval Ocean Systems Center
Attn: Code 753, P. H. Johnson
271 Catalina Boulevard
San Diego, California 92152

Donald E. Kirk
Professor and Chairman
Electronic Engineer, SP-304
Naval Postgraduate School
Monterey, California 93940

Naval Air Development Center
Attn: Code 01, Dr. R. K. Lobb
Johnsville
Warminster, Pennsylvania 18974

Naval Research Laboratory
Attn: Code 5270, B. D. McCombe
4555 Overlook Avenue, SW
Washington, D.C. 20375

Capt. R. B. Meeks
Naval Sea Systems Command, NC #3
2531 Jefferson Davis Highway
Arlington, Virginia 20362

Dr. H. J. Mueller
Naval Air Systems Command
Code 310, JP #1
1411 Jefferson Davis Highway
Arlington, Virginia 20360

Dr. J. H. Mills, Jr.
Naval Surface Weapons Center
Electronics Systems Department
Code DF
Dahlgren, Virginia 22448

Naval Ocean Systems Center
Attn: Code 702, H. T. Mortimer
271 Catalina Boulevard
San Diego, California 92152

Naval Air Development Center
Attn: Technical Library
Johnsville
Warminster, Pennsylvania 18974

Naval Ocean Systems Center
Attn: Technical Library
271 Catalina Boulevard
San Diego, California 92152

Naval Research Laboratory
Underwater Sound Reference Division
Technical Library
P. O. Box 8337
Orlando, Florida 32806

Naval Surface Weapons Center
Attn: Technical Library
Code DX-21
Dahlgren, Virginia 22448

Naval Surface Weapons Center
Attn: Technical Library
Building 1-330, Code WX-40
White Oak Laboratory
Silver Spring, Maryland 20910

Naval Training Equipment Center
Attn: Technical Library
Orlando, Florida 32813

Naval Undersea Center
Attn: Technical Library
San Diego, California 92152

Naval Underwater Systems Center
Attn: Technical Library
Newport, Rhode Island 02840

Office of Naval Research
Electronic and Solid State
Sciences Program (Code 427)
800 North Quincy Street
Arlington, Virginia 22217

Office of Naval Research
Mathematics Program (Code 432)
800 North Quincy Street
Arlington, Virginia 22217

Office of Naval Research
Naval Systems Division
Code 220/221
800 North Quincy Street
Arlington, Virginia 22217

Director
Office of Naval Research
New York Area Office
715 Broadway, 5th Floor
New York, New York 10003

Office of Naval Research
San Francisco Area Office
One Hallidie Plaza, Suite 601
San Francisco, California 94102

Director
Office of Naval Research Branch Office
495 Summer Street
Boston, Massachusetts 02210

Director
Office of Naval Research Branch Office
536 South Clark Street
Chicago, Illinois 60605

Director
Office of Naval Research Branch Office
1030 East Green Street
Pasadena, California 91101

Mr. H. R. Riedl
Naval Surface Weapons Center
Code WR-34
White Oak Laboratory
Silver Spring, Maryland 20910

Naval Air Development Center
Attn: Code 202, T. J. Shopple
Johnsville
Warminster, Pennsylvania 18974

Naval Research Laboratory
Attn: Code 5403, J. E. Shore
4555 Overlook Avenue, SW
Washington, D.C. 20375

A. L. Slafkovsky
Scientific Advisor
Headquarters Marine Corps
MC-RD-1, Arlington Annex
Washington, D.C. 20380

Harris B. Stone
Office of Research, Development,
Test and Evaluation
NOP-987
The Pentagon, Room 5D760
Washington, D.C. 20350

Mr. L. Sumney
Naval Electronics Systems Command
Code 3042, NC #1
2511 Jefferson Davis Highway
Arlington, Virginia 20360

David W. Taylor
Naval Ship Research and
Development Center
Code 522.1
Bethesda, Maryland 20084

Naval Research Laboratory
Attn: Code 4105, Dr. S. Teitler
4555 Overlook Avenue, SW
Washington, D.C. 20375

Lt. Cdr. John Turner
NAVMAT 0343
CP #5, Room 1044
2211 Jefferson Davis Highway
Arlington, Virginia 20360

Naval Ocean Systems Center
Attn: Code 746, H. H. Wieder
271 Catalina Boulevard
San Diego, California 92152

Dr. W. A. Von Winkle
Associate Technical Director for
Technology
Naval Underwater Systems Center
New London, Connecticut 06320

Dr. Gernot M. R. Winkler
Director, Time Service
US Naval Observatory
Massachusetts Ave. at 34th St., NW
Washington, D.C. 20390

Other Government Agencies

Dr. Howard W. Etzel
Deputy Director
Division of Materials Research
National Science Foundation
1800 G Street
Washington, D.C. 20550

Mr. J. C. French
National Bureau of Standards
Electronics Technology Division
Washington, D.C. 20234

Dr. Jay Harris
Program Director
Devices and Waves Program
National Science Foundation
1800 G Street
Washington, D.C. 20550

Los Alamos Scientific Laboratory
Attn: Reports Library
P. O. Box 1663
Los Alamos, New Mexico 87544

Dr. Dean Mitchell
Program Director
Solid-State Physics
Division of Materials Research
National Science Foundation
1800 G Street
Washington, D.C. 20550

Mr. F. C. Schwenk, RD-T
National Aeronautics and Space
Administration
Washington, D.C. 20546

M. Zane Thornton
Deputy Director, Institute for
Computer Sciences and Technology
National Bureau of Standards
Washington, D.C. 20234

Director
Stanford Electronics Laboratories
Stanford University
Stanford, California 94305

Nongovernment Agencies

Director
Columbia Radiation Laboratory
Columbia University
538 West 120th Street
New York, New York 10027

Officer in Charge
Carderock Laboratory
Code 18, G. H. Gleissner
David Taylor Naval Ship Research
and Development Center
Bethesda, Maryland 20084

Director
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

Dr. Roy F. Potter
3868 Talbot Street
San Diego, California 92106

Director of Laboratories
Division of Engineering and
Applied Physics
Harvard University
Pierce Hall
Cambridge, Massachusetts 02138

Director
Electronics Research Center
The University of Texas
Engineering-Science Bldg. 112
Austin, Texas 78712

Director
Electronics Research Laboratory
University of California
Berkeley, California 94720

Director
Electronics Sciences Laboratory
University of Southern California
Los Angeles, California 90007

Director
Microwave Research Institute
Polytechnic Institute of New York
333 Jay Street
Brooklyn, New York 11201

Director
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139